

**DEPARTAMENTO DE
TECNOLOGÍA ELECTRÓNICA Y DE LAS COMUNICACIONES**

**DISEÑO DE UN SISTEMA ROBÓTICO MULTISENSORIAL PARA LA
OPTIMIZACIÓN DE TÉCNICAS DE LOCALIZACIÓN Y BÚSQUEDA**

~ TESIS DOCTORAL ~

Autor: Julio Francisco Acosta Núñez

**(Ingeniero en Electrónica e Instrumentación, Universidad de las
Fuerzas Armadas – ESPE)**

Latacunga, Ecuador

Madrid, mayo de 2020.

Departamento: Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid (UAM), ESPAÑA

Tesis PhD: Diseño de un sistema robótico multisensorial para la optimización de técnicas de localización y búsqueda.

Autor: Julio Francisco Acosta Núñez
Ingeniero en Electrónica e Instrumentación
Universidad de las Fuerzas Armadas – ESPE
Latacunga, Ecuador

Director: Guillermo José González de Rivera Peces
Doctor Ingeniero de Telecomunicación
(Ingeniería por: Universidad Politécnica de Madrid)
Doctorado por: Universidad Autónoma de Madrid, España

Director: Víctor Hugo Andaluz Ortiz
Doctor Ingeniero en Electrónica y Control
(Ingeniería por: Escuela Politécnica Nacional, Quito, Ecuador)
Doctorado por: Universidad Nacional de San Juan, Argentina

Año: 2020

Comité:

La investigación descrita en esta tesis se llevó a cabo dentro del Grupo de Investigación Hardware & Control Technology Laboratory – HCTLab, en el Departamento de Tecnología Electrónica y de las Comunicaciones, Escuela Politécnica Superior, Universidad Autónoma de Madrid.

Derechos de autor © 2020 por Julio Francisco Acosta Núñez. Todos los derechos reservados. Ninguna parte de esta publicación puede reproducirse ni transmitirse de ninguna forma ni por ningún medio, ya sea electrónico o mecánico, incluyendo fotocopias, grabaciones o cualquier sistema de almacenamiento y recuperación de información, sin el permiso por escrito del autor. La Universidad Autónoma de Madrid tiene varios derechos para reproducir y distribuir electrónicamente este documento.

A ti Virgen María.....Totus Tuus

Agradecimientos.

Después de permanecer en la academia tanto tiempo, he llegado a comprender de mi propia pequeñez, ya que cada día hay más gente brillante que desarrolla trabajos brillantes, muchos de ellos nos han cambiado la vida a todos; entonces no se puede pretender ser una única luz que brilla, pero si reconoces que el camino persiste, nada se mantiene estático y somos únicamente eslabones de la gran cadena que avanza y nada la detiene.

Muchas veces he considerado que esta meta que me propuse al terminar los estudios del máster, era tan alta que jamás podría alcanzarla, pero lo he podido hacer, por supuesto que sería aberrante y muy desconsiderado pretender que lo he hecho solo; más bien sin ayuda, creo que sí, jamás la hubiera podido alcanzar; es por eso que es justo agradecer a todos quienes fueron esa ayuda en todo el tiempo, que no ha sido corto, que ha transcurrido desde que decidí emprender esta campaña. En primer lugar, agradecer a ese ser inmaterial que me acompañó y habló todos los días y las noches, especialmente en los de frustración y angustia, que me hicieron reconocer su presencia verdadera a mi lado y su amor, a ese Dios uno y trino que nunca me faltó y que es y será motivo de mi felicidad eterna, Él fue quien ha puesto a mi lado y en mi camino a las demás personas que me empujaron y animaron de una u otra forma a seguir, perseverar y luchar; Mi familia, ese pilar, esa roca donde se fundamenta el amor, la responsabilidad y la obligación; Mi esposa Lucy y mis dos hijos queridos, Alejandro y Ariana que siempre estuvieron dando la palabra de aliento adecuada y muchas veces sin entender, el ofrecerme su ayuda y colaboración, sin saber que su afecto era el insospechado combustible contra la fatiga y el desaliento.

Fuera de mi hogar estoy muy agradecido con Guillermo González de Rivera y Javier Garrido, del grupo de investigación HCTLab de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid, lejos de mi casa y mi país en las instalaciones del laboratorio me hicieron sentir siempre como en casa y parte de su familia, además que fueron de mucha ayuda académicamente y sobre todo por su experiencia.

Pero sin lugar a dudas mi reconocimiento especial a Víctor Andaluz, ese joven doctor del cual he aprendido tanto, en la ciencia y en la humildad y sencillez, gracias por su invalorable contribución a este proyecto.

Finalmente, a la academia, representada por mis dos instituciones formadoras y auspicientes: La Universidad Autónoma de Madrid en España y la Universidad de las Fuerzas Armadas – ESPE en Ecuador, que han sido, las responsables y gestoras de esta investidura y este honor.

Julio Acosta N.

mayo 2020.

Resumen.

El presente trabajo doctoral, nace de la necesidad de implementar sistemas autónomos, robóticos; que permitan resolver algunas de las interrogantes no resueltas en trabajos anteriores a los estudios de doctorado; específicamente no resueltos, en la investigación del trabajo de fin de máster del programa de Máster Universitario en Investigación e Innovación en Tecnologías de la Información y las Comunicaciones; en este trabajo se desarrolló un sistema robótico simple y de bajas prestaciones, al cual se le proyectaba conectar una nariz electrónica para que sirviera como elemento de búsqueda de fuentes de odorantes para aplicaciones específicas. Lamentablemente y por la imposibilidad de contar hasta el último día de plazo de presentación con la nariz electrónica, se decide hacer un sistema que simule de la forma más cercana a dichas características. Se hacen investigaciones y se determina que la luz infrarroja tiene una “pluma” de dispersión muy parecida a la de olores característicos de las cosas, es así entonces que se reemplaza a la nariz inteligente, por un arreglo de sensores infrarrojos, que se colocan en el robot para que sean capaces de detectar la posición y concentración de una fuente de luz infrarroja.

El reemplazo de este sensor por el de la nariz inteligente, no conlleva variación en todos los demás aspectos del sistema y el robot, ya que sus algoritmos y funciones (hardware y software) de movimiento, detección, seguimiento de fuente de un fenómeno natural (originalmente olor y reemplazado con luz) eran exactamente los mismos, entonces se implementa este sistema consiguiendo plenamente los resultados esperados.

Este éxito en la consecución de resultados, y principalmente los aspectos que introducían error en éstos, hacen que se generen interrogantes que se puedan resolver con investigaciones más amplias en lo que tienen que ver al conjunto que dispone un robot en hardware y software; además y principalmente, con el intercambio de los sensores, se nota que se pueden conectar múltiples sensores a un robot de tal manera que al poseer múltiples sentidos, pueda realizar múltiples tareas, es decir, se podría rápidamente deducir que un robot multi sensor podría realizar cualquier tarea que se encargue.

Se trabaja entonces, con la idea de una plataforma multisensorial, es decir un sistema de hardware que permita conectar a él cualquier sensor de variable dinámica existente; esto no tiene mucho sentido si se analiza solamente la capa física, es decir la interconectividad; no sirve de mucho (pero por definición ya es una ventaja) que cualquier sensor se conecte en la plataforma, sino que se debe operar adecuadamente, es decir la plataforma de hardware debe entender lo que estas señales de entrada significan, lo que es lo mismo, interoperabilidad, de tal manera que el control de todas estas variables sea efectivo dentro de la plataforma; esto se logra muchísimo mejor con la estandarización de las señales, procedimiento que ayuda a que, sea de

la naturaleza que sea la variable o el sensor, sus señales sean estandarizadas en capa física y sean reconocidas, interpretadas y operadas en un hardware específico o mejor en cualquier hardware. Finalmente, lo mejor de este aspecto estandarizado es obtener la intercambiabilidad, es decir en esa misma entrada física colocar sensores de distintas naturaleza y características de sentido o variable, y sean intercambiados fácilmente entre ellos.

El siguiente aspecto que se estudia es el control (mediante hardware) de las aplicaciones que se realizarán una vez conectados e interpretados todos los sensores; originalmente se había utilizado una tarjeta de desarrollo embebida para realizar todas las tareas que desarrollaba el robot: la medición del fenómeno que desencadena la tareas (olor / luz), como elemento principal; luego el control del movimiento, que consistía únicamente en una aplicación de señales PWM proporcionales al movimiento de cada rueda, a los motores del robot y la medición de dichos movimientos a través de encoders colocados en sus ruedas; esto era suficiente para la aplicación original, es decir, se desarrolló un algoritmo que dirija al robot hacia una fuente de odorante (luz infrarroja) una vez detectada su intensidad desde un umbral mínimo; pero se extendió esta aplicación a la de un sistema de supervisión de estas tareas desde un ordenador; esto demanda ya que el hardware del robot (tarjeta de desarrollo embebida) tenga además la posibilidad de comunicarse con este ordenador a través de medios inalámbricos, más aún si se habla de términos de colaboración entre robots, que significa que esta red deba tener no solamente un entorno maestro-esclavo sino un entorno de red completo con muchos hosts.

En este sentido, el primer escollo del trabajo fue el uso de esta plataforma inteligente que interprete las señales de los sensores y los lleve a un algoritmo de actuación; esta situación casi no tuvo relevancia en el trabajo o versión original ya que solamente se controlaba el movimiento de las ruedas del robot según la presencia o no de una porción de la pluma de odorante y se hacía mediante el uso de las señales análogas de la placa; pero al momento de agregarle funcionalidades se hizo poco probable que se pudiera realizar con esa o algunas alternativas de plataformas embebidas; la razón: los algoritmos de programación, pues estos ya no eran capaces de desarrollarse en programas sencillos, y con estructuras fáciles, sino que necesitan de lenguajes de alto nivel y con orientación matemática para poder desarrollar los algoritmos en base a los modelos matemáticos; eso significó también un giro importante en el diseño y construcción de los robots de este trabajo con respecto al diseño original; pues se necesitó realizar un diseño mecánico específico para cada robot orientado a cada aplicación, tomando muy en cuenta sus medidas, pesos, movilidad, etc., esto pues los algoritmos toman en consideración todos estos aspectos; además, en las aplicaciones ahora se introducen varios robots en tareas colaborativas, lo que hace necesario también el estudio y diseño de una red de comunicaciones inalámbricas, que no presenten latencias tan grandes que los algoritmos no puedan medir y controlar, desde estas tramas de comunicaciones y en tiempo real, las posiciones

y movimientos de los robots. Finalmente se necesita también implementar un sistema de supervisión desde una PC para aplicar conceptos como servicios de sistemas de control distribuido, SCADA y telemando en características remotas. Es por todo esto que la solución final es colocar como elemento principal o inteligencia central de las aplicaciones a una computadora que lo desarrolle todo y para efectos de disminuir su carga computacional y escalar los algoritmos de control, los controladores del movimiento residan en cada robot y se realice con tarjetas embebidas.

Se ha realizado también un diseño mecánico detallado para poder minimizar los errores que producen las características físicas de un robot y de su ambiente circundante, y se han adaptado a estos diseños, los diseños de los controladores, para producir los mejores resultados, además de que se pueda producir un sistema que introduzca confiabilidad en las aplicaciones.

Finalmente se ha desarrollado un sistema de software que también puede conectarse en tiempo real a la plataforma completa, a fin de que se realicen eventos de supervisión y teleoperación remotas en los robots, en forma individual, o actuar sobre el sistema completo; en este aplicativo, se han desarrollado sistemas de comunicaciones entre API's del sistema operativo para reducir las latencias que se pueda introducir también, y ambientes de realidad virtual, que se pueden conectar con el conjunto enorme de sistemas hápticos existentes en la actualidad.

Para concluir con el trabajo se hicieron desarrollos que a través de pruebas demostraron la eficacia de los controladores sobre las plataformas diseñadas y construidas para el efecto, además de presentar resultados muy satisfactorios de las respuestas de éstos.

Con todo este trabajo, esta tesis doctoral demuestra que el diseño e implementación de robots móviles en ambientes particulares o colaborativos responden muy satisfactoriamente a las tareas de localización y búsqueda de ellos mismos y otros en tareas colaborativas; se usan como herramientas de comprobación herramientas poderosas para tareas prácticas y novedosas, se desarrollaron también sistemas eficientes a la hora de reducir errores de odometría en plataformas móviles y de consumo de energía en tareas de movimiento y manipulación; finalmente el uso de una red eficiente y un sistema de monitores de adecuadas características a los desarrollos actuales.

A continuación, se presenta el desarrollo de todo lo mencionado en estos párrafos anteriores.

Contenido

1	INTRODUCCIÓN.....	- 1 -
1.1	Motivación de la tesis	- 4 -
1.2	Contribuciones de la tesis	- 5 -
1.3	Contribuciones Científicas.	- 5 -
1.3.1	Revistas Internacionales con factor de impacto.....	- 5 -
1.3.2	Conferencias Internacionales.	- 6 -
1.4	Organización de la tesis.....	- 6 -
2	ESTUDIO DEL ESTADO DEL ARTE.....	- 9 -
2.1	PRIMER EJE: ROBOTS MULTISENSORIALES.....	- 9 -
2.2	SEGUNDO EJE: ROBOTS MOVILES COLABORATIVOS	- 13 -
2.3	TERCER EJE: REDUCCION DEL CONSUMO DE ENERGIA EN ROBOTS MOVILES	- 18 -
2.4	CUARTO EJE: CORRECCION DE ERRORES DE ODOMETRIA PARA ROBOTS MOVILES.	- 22 -
3	DISEÑO Y CONSTRUCCIÓN DE LAS PLATAFORMAS ROBÓTICAS MÓVILES MULTISENSORIALES.	- 29 -
3.1	Construcción de las plataformas robóticas para aplicación.....	- 36 -
3.1.1	Robots de tipo Uniciclo.	- 36 -
3.1.2	Análisis estructural.	- 42 -
3.1.3	Diseño del sistema eléctrico y electrónico.	- 44 -
3.1.4	Robots de Tipo Omnidireccional.....	- 52 -
3.1.5	Diseño del sistema electrónico y de control.....	- 58 -
3.1.6	Construcción de la plataforma OMNIBEE.....	- 63 -
3.1.7	Construcción del Manipulador móvil.....	- 69 -
4	MODELACIÓN MATEMÁTICA DE ROBOTS MÓVILES Y MANIPULADORES MÓVILES..	- 79 -

4.1	Introducción.....	- 79 -
4.2	Modelos matemáticos de sistemas de control de robots móviles.	- 81 -
4.2.1	Modelo Cinemático de los robots móviles, generalidades.	- 82 -
4.2.2	Modelo dinámico de los robots móviles, generalidades.	- 83 -
4.3	Modelos cinemáticos para las plataformas de aplicación diseñadas y construidas..	- 84 -
4.3.1	Robots tipo Uniciclo.....	- 84 -
4.3.2	Robots Omnidireccionales.	- 86 -
4.3.3	Robots Manipuladores.	- 88 -
4.3.4	Robots Manipuladores Móviles	- 90 -
4.4	Modelos dinámicos para las plataformas de aplicación diseñadas y construidas. ...	- 92 -
4.4.1	Robots Uniciclo.....	- 92 -
4.4.2	Robot Omnidireccional.	- 94 -
4.4.3	Robot Manipulador.	- 96 -
4.4.4	Robot Manipulador móvil.	- 100 -
5	DISEÑO E IMPLEMENTACIÓN DE LA RED INALÁMBRICA PARA COMUNICACIÓN ENTRE ESTACIONES ROBÓTICAS INTELIGENTES PARA TRABAJOS AUTÓNOMOS O COOPERATIVOS Y DE LA PLATAFORMA DE SUPERVISIÓN DEL SISTEMA DE ROBOTS MÓVILES.	- 113 -
5.1	Introducción.....	- 113 -
5.2	Sistemas por ultrasonidos.	- 113 -
5.3	Sistemas Infrarrojos.....	- 114 -
5.4	Sistemas FSO.....	- 115 -
5.5	Sistemas de Radio.....	- 115 -
5.5.1	Sistema Bluetooth.....	- 116 -
5.5.2	Sistemas WiFi.....	- 118 -
5.5.3	Sistema ZigBee.	- 121 -
5.5.4	Sistemas de Telefonía móvil.	- 123 -
5.5.5	Comparación entre los sistemas infrarrojos, FSO y Ultrasonidos.	- 126 -
5.5.6	Comparación entre Bluetooth, WiFi y ZigBee.	- 126 -

5.5.7	Comparación entre sistemas GSM; GPRS y 3, 4, 5G	- 127 -
5.6	Desarrollo de la red ZigBee en los robots contruidos.	- 129 -
5.6.1	Tipos de nodos a configurar:.....	- 130 -
5.7	Diseño de la plataforma de supervisión.	- 132 -
5.8	Creación de la interfase de supervisión 3D.	- 134 -
5.8.1	Importación de un modelo 3D.	- 134 -
5.8.2	Movimiento de rotación.....	- 136 -
5.8.3	Movimiento de Rigging.	- 137 -
5.8.4	Ambiente Unity en el objeto de simulación.	- 138 -
5.8.5	Interacción del ambiente con las aplicaciones.	- 140 -
5.8.6	Conexión entre Unity y MATLAB (y otras aplicaciones).....	- 141 -
6	APLICACIONES, PRUEBAS Y RESULTADOS.	- 145 -
6.1	PRIMERA APLICACIÓN: Reducción de los errores de odometría en robots móviles.	- 146 -
6.1.1	Algoritmo propuesto.	- 149 -
6.1.2	Implementación.....	- 158 -
6.1.3	Resultados experimentales.	- 162 -
6.2	SEGUNDA APLICACIÓN: Control heterogéneo de robots considerando objetivos secundarios.	- 167 -
6.2.1	Esquema de control Multicapa.	- 168 -
6.2.2	Transformación Cinemática.	- 170 -
6.2.3	Diseño de los controladores.	- 173 -
6.2.4	Control de Formación y seguimiento de caminos.....	- 176 -
6.2.5	Control Cinemático y compensación dinámica en cada robot.....	- 182 -
6.2.6	La instrumentación en los robots.....	- 184 -
6.2.7	Resultados experimentales.	- 186 -
6.3	TERCERA APLICACIÓN: Desarrollo de la aplicación de ahorro de energía para tereas de manipuladores móviles.	- 193 -
6.3.1	Propuesta de ahorro de energía.....	- 194 -

6.3.2	Modelo del manipulador móvil.....	- 195 -
6.3.3	Diseño del controlador.....	- 198 -
6.3.4	Resultados experimentales.....	- 210 -
7	CONCLUSIONES Y TRABAJO FUTURO.....	- 225 -
7.1	Conclusiones.....	- 225 -
7.2	Trabajos futuros.....	- 227 -
7.3	Contribuciones Científicas.....	- 228 -
7.3.1	Revistas Internacionales con factor de impacto.....	- 229 -
7.3.2	Conferencias Internacionales.....	- 229 -
	Referencias	- 231 -

Capítulo I

1 INTRODUCCIÓN.

Antes de que los primeros mecanismos robotizados, es decir que pueden funcionar por sí mismos, fueran contruidos por relojeros en la Europa del siglo XVIII, en muchas de las culturas antiguas ya se fabricaban mecanismos de diversa índole, que operaban de acuerdo a leyes de la física, pero con invenciones hasta cierto punto inverosímiles que les comunicaban movimiento y otras particularidades. Por todas estas máquinas y a partir de muchas épocas y descubrimientos, ha existido una fascinación de los seres humanos por construir artefactos que, a más de divertirle, fueren de ayuda a sus muchas veces limitadas capacidades de movilidad, fuerza y destreza. Es indudable comprobar que, a través del tiempo, los desarrollos científicos y tecnológicos han hecho que esta fascinación pueda llevarse a cabo, y que tenga una relación directa: más tecnología, más invenciones; así como también ha tenido tal interés cuanto más por el carácter interdisciplinario de la robótica.

Hace más de 20 años, cuando el desarrollo de las ciencias era cada vez más explícito, las técnicas para construir robots se empezaron a diversificar, pues la naturaleza de estos autómatas se volvió muy diversa, así como los ambientes en los que podían trabajar. Es verdad que toda esta evolución es producto de diversas raíces, así, la mecánica, la electrónica, el control, la cibernética, e incluso las ciencias sociales, con la inclusión de los autómatas en el entorno del quehacer y las relaciones sociales de los humanos, etc. Todo esto ha hecho que los campos de investigación se hayan multiplicado por decir lo menos.

En cuanto al punto de vista meramente tecnológico, los avances que ahora se perciben en robótica, han evolucionado de la mano de la misma evolución de la electrónica especialmente y la computación; así es como, el mayor foco de atención se produce cuando estas máquinas comienzan a ser capaces de reemplazar a los seres humanos en tareas comunes y aventajarlo en situaciones que requieren de exactitud y ejecución de grandes lotes de procesos sistemáticos. Esto demanda, de más estudios diversificados de modelización, planificación y control.

Visto lo anterior, la robótica se introduce agresivamente en la industria de fabricación relegando en algunos casos al hombre por su versatilidad y carencia de errores en las aplicaciones, lo que lleva suponer, y por ende a investigar otras áreas de aplicación.

Las primeras y exitosas aplicaciones de robots en las industrias se focalizan en los brazos robóticos, que en principio son estáticos y que únicamente se mueven en torno de la aplicación de sus (1,2,3, etc.) extremos articulados. En cambio, los robots móviles son capaces de desplazarse por entornos muchos más grandes e incluso desconocidos. Es ahí en donde entran

de forma directa los avances tecnológicos, pues originalmente estos aparatos carecían de mecanismos que les permitiera, en primer lugar, el movimiento, luego la autonomía y finalmente la forma de reconocer lo desconocido, y actuar sobre ello. No es hasta la presente era, es decir el salto del siglo a lo mejor, cuando se dispone de todo tipo de sensores que le permiten a un robot interpretar absolutamente todo lo que hay en el ambiente que le rodea: si hay obstáculos, si se desplaza por suelos rugosos, en ambientes dentro o fuera del control, si su entorno es un ambiente amigable de mucho oxígeno, polución, gas o intrínsecamente seguro, la luminosidad que encontrará, si deberá saltar, volar, sumergirse , etc. Esto obliga a que pueda enfrentarse de forma efectiva a todas estas alternativas, para ello se hacen necesarias más técnicas de control nuevos actuadores, y elementos de interacción especiales. Hay ahora entonces robots que se mueven al mismo tiempo que ven, oyen, olfatean, degustan y palpan, es decir como un ser humano, y además actúan sobre cada característica descrita con solvencia y rapidez.

En lo que respecta a la robótica móvil específicamente, los diseños de las aplicaciones dependen de varios factores muy importantes: en primer lugar su característica de movimiento, definido por sus características mecánicas o dinámicas, que originalmente se puede pensar no influyen directamente en su comportamiento al ser un elemento controlado electrónicamente y de forma inteligente; al contrario, mientras mejor establecidas sean sus características mecánicas y cinemáticas, mejor se podrán adaptar los mecanismos de control a realizar las tareas casi sin errores, lo que en el argot de la electrónica de control se conoce como disminuir la carga del proceso.

Por otra parte, y, en segundo lugar, están estos elementos de control electrónico inteligente, que no son simplemente dispositivos semiconductores programables, sino sistemas que a más de permitir desarrollar en ellos algoritmos muy especializados y complejos, permiten también comunicar desde o hacia, las variables y factores de desarrollo, al mismo tiempo que registrar datos, visualizarlos, estandarizarlos, etc. Esta característica hace que se requieran eventualmente sistemas muy inteligentes o principalmente ordenadores de varios tamaños y prestaciones exigentes (cumplimiento de normas NEMA, IP, etc.).

En tercer lugar, la ciencia, que permita desarrollar los algoritmos de control basados en postulaciones y comprobaciones matemáticas, que llevan a producir errores muy pequeños o a controlarlos, es determinante; pues no solamente consiste en darle energía a un motor para que gire en conjunto con todo lo demás, sino saber cuánta energía suministrarle y en qué condiciones y aplicaciones. Y, finalmente, la autonomía energética que debe tener una máquina de este tipo y como se la administrará generará/regenerará, en fin.

Este trabajo nace de la necesidad de encontrar sistemas que de una u otra forma resuelvan las limitaciones que se encontraron al realizar el Trabajo de Fin de Máster en el programa de Máster oficial en Investigación e Innovación en Tecnologías de la Información y las Comunicaciones, al finalizar el año 2013. En este TFM, se plantea la necesidad de montar una nariz electrónica sobre un robot móvil que tenga la capacidad de interactuar en un ambiente conocido para la búsqueda de olores característicos o una fuente cualquiera de odorante.

En este sentido se plantean muchas interrogantes, desde el diseño mismo de las arquitecturas robóticas que se pueden utilizar en aplicaciones como ésta y en muchas otras más, hasta la conceptualización, diseño y elaboración de algoritmos inteligentes y sistemas que permitan realizar tareas complejas con robots móviles, especialmente tareas colaborativas en entornos estructurados o parcialmente estructurados, en los cuales intervengan robots de distintas características y orientaciones de aplicación.

El primer enfoque en la búsqueda de las soluciones, se centra entonces en el hardware; se deben construir sistemas robóticos que permitan realizar las operaciones, aplicaciones y tareas fundamentales; para esto es determinante definir dos características completamente diferenciadas: a) su diseño mecánico, que permita aplicar sus características particulares cinemáticas y dinámicas perfectamente establecidas, en la tarea que se les vaya a ordenar; y, b) la electrónica de control y las comunicaciones, que permitirán el comportamiento inteligente de dicha plataforma, justamente considerando sus parámetros cinemáticos y dinámicos, de forma que se pueda actuar sobre ellos modificándolos para que se puedan conseguir los fines perseguidos por cada robot, y en un conjunto que permita desarrollar además, un trabajo colaborativo eficaz, tomando como consideración principal la autonomía energética en las tareas encomendadas, es decir un análisis del consumo y ahorro de energía en sus tareas.

Entonces es construir un robot, o una familia de robots que en principio perciban de todo, es decir se les monte una cantidad muy grande de sensores y medidores, que esto les permita reducir errores de localización de sí mismos y de los demás (en entorno colaborativo), que puedan desplazarse eficientemente siguiendo trayectorias ordenadas por un sistema superior o las condiciones que ellos mismos puedan leer, que consuman la menor cantidad de energía en las tareas desarrolladas de tal forma que pueda administrarse adecuadamente su energía y finalmente que trabajen con diversos tipos de robots en tareas de cooperación, para lo que será determinante la inclusión de un sistema de comunicaciones que no retarde ni entorpezca las operaciones sino sea casi transparente en su operación.

Este trabajo contempla todo esto y se han realizado aplicaciones que permiten corroborar todo lo planteado.

1.1 Motivación de la tesis

Tomando como referencia, los trabajos que se han podido analizar y resumir específicamente en el Capítulo II de esta memoria, la construcción de plataformas robóticas que puedan ser utilizadas para que desarrollen aplicaciones específicas de manera efectiva; la integración de la inmensa cantidad de sensores existentes en la actualidad, la facilidad de implementar estrategias computacionales sobre estos robots de distintas naturaleza, ámbitos y ambientes de trabajo; exige no solamente el diseño de un hardware que sea efectivo al momento de la aplicación, sino de más elementos que minimicen la introducción de errores en los procedimientos ya inteligentes de aplicación; es por esta razón que en este trabajo de tesis se ha emprendido, no solamente el diseño electrónico de las plataformas, sino atacar también a las fuentes de error, que son sus características mecánicas y de movimiento; al diseñar los robots en sus características mecánicas, también se consigue la ventaja adicional de introducir estas características de diseño dentro de los algoritmos de operación, es por eso que también surge la necesidad de modelar de forma matemática sus características cinemáticas y dinámicas en los ambientes de probable actuación.

En el campo meramente del hardware, los diseños también deben proporcionar sistemas que introduzcan menor latencia en la operación del conjunto, sean capaces de inhabilitarse mientras no están en operación y consuman muy poca potencia, tanto en operación como fuera de ella; es así que se selecciona y comprueban los rendimientos en distintas plataformas de hardware para definir sus mejores singularidades; estos diseños deben ser capaces de incluir de forma eficaz la mayoría de sensores disponibles en el mercado, sea únicamente para mantenerlos como elementos de reserva, con o sin conexión o activos todo el tiempo, esta característica le dará su dimensión de multi-sensorialidad. Finalmente, las estrategias de operación, plasmadas en programas computacionales complejos, deben ser las que asuman lo que el hardware y la mecánica ya no lo pueden hacer, es decir, definir un conjunto de pasos y procedimientos operativos brillantes que lleven a la consecución de los objetivos que la parte física no puede llegar a hacerlo por sus limitaciones existentes de por sí.

En este trabajo se propone todo lo mencionado: los diseños mecánicos y electrónicos más adecuados para ejecutar las aplicaciones; la aplicación de las estrategias que se consideran más novedosas, que lleven a cumplir objetivos de trabajo colaborativo de conjuntos de robots, reducción de errores de posicionamiento y reducción de consumos de energía, además de darle la posibilidad al sistema completo de regularse de forma autónoma o de forma supervisada por el ser humano, dentro de un ambiente automático e inalámbrico que considere las respuestas más rápidas y efectivas.

1.2 Contribuciones de la tesis

La contribución principal de esta tesis se basa en el análisis e integración de tres elementos básicos de robots móviles: sus diseños mecánicos y sus características e influencia de los modelos cinemáticos y dinámicos que poseen éstos, en la aplicación de operaciones individuales y cooperativas, para el cumplimiento de tareas; la integración efectiva y racional de la mayor cantidad de sensores sobre una plataforma básica o extensible de hardware, que permita múltiples mediciones simultáneas o alternadas; y la elaboración de estrategias que combinen las dos características anteriores para la aplicación de tareas que incluyan; trabajo colaborativo, búsqueda y localización de robots, en singular o en colaboración, reducción de errores de odometría, reducción del consumo de energía en la elaboración de las tareas; todo ello basado en un marco de operación autónoma o supervisada, que incorpore técnicas de manejo de la intuitividad de la información y teleoperación desde un centro de acopio (MTU) computacional, administrado por comunicaciones inalámbricas entre todos los actores del sistema, que incluyan alcance y facilidad de operación.

1.3 Contribuciones Científicas.

Los desarrollos de esta tesis han dado lugar a tres publicaciones en revistas internacionales (una de ellas está en revisión) y cuatro publicaciones en conferencias internacionales, estos trabajos los cito a continuación:

1.3.1 *Revistas Internacionales con factor de impacto.*

- “Multirobot heterogeneous control considering secondary objectives”, Acosta, J. F., de Rivera, G. G., Andaluz, V. H., & Garrido, J. (2019), *Sensors* (Switzerland), 19(20) doi:10.3390/s19204367.
- “Energy-Saver Mobile Manipulator Based on Numerical Methods”, Acosta Núñez, J.F.; Andaluz Ortiz, V.H.; González-de-Rivera Peces, G.; Garrido Salas, J. (2019) *Electronics* (Switzerland) 2019, 8(10), 1100; <https://doi.org/10.3390/electronics8101100>.
- “Redundant system for the correction of odometry errors for mobile unicycle and omnidirectional robots using artificial vision and genetic algorithms”, Acosta Núñez, J.F.; Andaluz Ortiz, V.H.; González-de-Rivera Peces, G.; Garrido Salas, J. (2019) *Electronics* (Switzerland) (ARTICULO EN REVISION)

1.3.2 Conferencias Internacionales.

- “3-D path planning using subatomic particles and feynman diagrams.”, Acosta, J. F., Andaluz, V. H., Naranjo, M. X., Molina, J. I., Santana, A. G., Topa, A. O., & Erazo, G. (2019). Paper presented at the Proceedings of the 3rd World Conference on Smart Trends in Systems, Security and Sustainability, WorldS4 2019, 368-375. doi:10.1109/WorldS4.2019.8903957
- “Path Planning Based in Algorithm Rapidly-Exploring Random Tree RRT”, Asqui, Leonardo; Andaluz, Víctor; Sánchez, Jorge; Acosta, Julio, Advanced Science Letters, Volume 24, Number 11, November 2018, pp. 8831-8836(6), American Scientific Publishers DOI: <https://doi.org/10.1166/asl.2018.12356>.
- “Autonomous assistance system for people with amyotrophic lateral sclerosis”, Santana G., A., Ortiz C., O., Acosta, J. F., & Andaluz, V. H. (2017). Lecture Notes in Electrical Engineering Volume 449, 2017, Pages 267-277, 7th International Conference on IT Convergence and Security, ICITCS 2017; Seoul; South Korea; 25 September 2017 through 28 September 2017; Code 193979 doi:10.1007/978-981-10-6451-7_32
- “Implementing a multisensory robotic platform for building collaborative monitored robots.”, Acosta, J. F., Glez-de-Rivera, G., & Garrido, J. (2014). Paper presented at the Proceedings of the 2014 29th Conference on Design of Circuits and Integrated Systems, DCIS 2014, doi:10.1109/DCIS.2014.7035592.

1.4 Organización de la tesis.

Al estar enfocado este trabajo de tesis a la construcción del hardware, software, desarrollo de estrategias y aplicaciones, para robots móviles en trabajo individual o cooperativos y comprobar estas aplicaciones con sistemas que organicen y efectúen tareas de colaboración en el transporte de objetos de diferente configuración, reducción de errores de odometría y reducción del consumo de potencia en el sistema global, está organizado de la siguiente manera:

- Capítulo 1: **INTRODUCCIÓN:** Se proporcionan las nociones básicas generales de lo que el trabajo aborda y de cómo se ha desarrollado los hitos específicos durante estos años; qué ha motivado su realización y cuáles han sido las contribuciones que se han podido aportar como resultados de las investigaciones y desarrollos hechos.
- Capítulo 2: **ESTUDIO DEL ESTADO DEL ARTE:** Se analizan en cuatro ejes fundamentales los trabajos más actuales y relevantes, que constituyen los avances globales en torno a cada uno de esos ejes, y de una u otra forma influyen en los desarrollos de esta tesis, a pesar de no estar directamente basada en uno o varios de ellos.

- **Capítulo 3: DISEÑO Y CONSTRUCCIÓN DE LAS PLATAFORMAS ROBÓTICAS MÓVILES MULTISENSORIALES:** Se presentan todas las instancias de diseño y construcción del hardware de todos los robots que se han empleado en las implementaciones de las aplicaciones mencionadas, estos diseños se realizan en cuanto a la parte mecánica y la electrónica, se detallan todos los elementos diseñados para su mejor operación.
- **Capítulo 4: MODELACIÓN MATEMÁTICA DE ROBOTS MÓVILES Y MANIPULADORES MÓVILES:** Se presenta la modelación cinemática y dinámica de los robots contruidos, como mecanismo que introduzca a la elaboración de estrategias y principalmente algoritmos de control, que sean la base de controladores electrónicos contruidos sobre estos robots para consecución de tareas específicas dentro de las aplicaciones.
- **Capítulo 5: DISEÑO E IMPLEMENTACIÓN DE LA RED INALÁMBRICA PARA COMUNICACIÓN ENTRE ESTACIONES ROBÓTICAS INTELIGENTES PARA TRABAJOS AUTÓNOMOS O COOPERATIVOS Y DE LA PLATAFORMA DE SUPERVISIÓN DEL SISTEMA DE ROBOTS MÓVILES:** Se presenta, como complemento a los sistemas robóticos, un estudio y diseño de la red de comunicaciones para el desarrollo del ambiente individual, y el cooperativo entre los robots contruidos y en aplicación, así también se presenta el desarrollo del sistema de supervisión, y teleoperación.
- **Capítulo 6: APLICACIONES, PRUEBAS Y RESULTADOS:** Se presentan las aplicaciones que conducen a cumplir los objetivos del trabajo planteado en el proyecto, y a la comprobación a través de ellas, de la operación de los robots en el cumplimiento de las tareas asignadas en cada aplicación: en primer lugar en la corrección de errores de odometría a través de un sistema redundante; la colaboración, y localización de robots en formación, para seguir una trayectoria determinada y finalmente la reducción del consumo de energía en tareas asignadas a un manipulador móvil.
- **Capítulo 7: CONCLUSIONES Y TRABAJO FUTURO:** Presenta las conclusiones a las que se ha llegado en el desarrollo de la tesis y la propuesta de trabajos futuros que se basen en la tesis desarrollada.

Capítulo II

2 ESTUDIO DEL ESTADO DEL ARTE.

Por todo lo mencionado en el Capítulo I, y por la amplia diversificación de ciencias y técnicas que tiene actualmente la robótica, y en especial la robótica móvil, sería demasiado extenso realizar un estudio del estado del arte de la robótica móvil, por lo que se han considerado cuatro ejes o aspectos fundamentales de estudio, que son los mismos sobre los cuales se actuará principalmente en esta tesis

Estas son las razones por las cuales este capítulo se ha organizado en cuatro aspectos, considerados los ejes básicos sobre los cuales se desarrolla esta tesis; estos son: la conceptualización y operación de sistemas de robots multisensoriales, los sistemas colaborativos entre robots móviles, la reducción de la energía consumida por robots móviles en tareas individuales o colectivas (en colaboración) y finalmente la corrección de errores de odometría, como técnica fundamental de localización de la posición de una o más plataformas robóticas en ambientes estructurados y no estructurados.

En este contexto se ha realizado un estudio del estado del arte de los trabajos relacionados con cada uno de estos ejes mencionados, considerando la tecnología más vanguardista, para poder observar posteriormente las coincidencias y diferencias; ventajas y desventajas con los elementos que se han construido tanto en hardware como en software en esta tesis.

2.1 PRIMER EJE: ROBOTS MULTISENSORIALES.

El uso de muchos sensores para efectuar aplicaciones especializadas con robots móviles no es nuevo y ha sido desarrollado por varios trabajos, En [1] se establece la necesidad de mantener los modelos del entorno en donde los robots van a trabajar y el conocimiento de la posición y tamaño de los objetos que va a encontrar en el camino de la aplicación, obviamente esto demanda de un sistema básico de sensores montados sobre ellos, o de un sistema sofisticado; eso está determinado por la aplicación que efectúe; aquí se han introducido cámaras RGB, perfiladores (sensores) de láser y sensores táctiles para reproducir adecuadamente el modelo necesario para la aplicación. Es interesante ver que en etapa de sensado, dependiendo de la capacidad y cantidad de puntos de detección elegirán entre el Kinetic (dispositivo introducido originalmente como controlador de video juegos que permite a su operador, controlar e

interactuar con la consola sin tener contacto físico con ella) y el sensor de laser; con ello determinan los espacios vacíos e inicialmente aquellos espacios que pueden presentar la presencia de un objeto a considerarse para construir el mapa.

Para detectar los vóxeles azules, amarillos, plomos y blancos que representan cada uno un límite de superficie encontrado, se cruza la información entre los tres sensores que se usan de forma itinerante en el trayecto y de ahí se empieza a construir un mapa.

Es necesario también definir que una plataforma simultáneamente a lo dicho, puede hacer uso de sensores SLAM [2] que irán comparando sus referencias instantáneas con la cartografía previamente obtenida, para definir instantáneamente el camino por donde el robot puede seguir. La técnica para determinar los puntos coincidentes entre los mapas prefabricados y la respuesta de LIDAR (Laser Imaging Detection and Ranging) es montar un filtro de partículas que elimina las partículas que producen inconsistencias.

Allí se realiza un método de localización basado en un filtro de Kalman para un vehículo. Los datos de entrada consisten en lecturas LIDAR, GPS o mediciones de brújula. El resultado es una localización de vehículos, que se basa en un mapa ya existente. La plataforma está lo suficientemente dispuesta para aceptar cualquiera de estos tres tipos de sensores, por cualquier vía de comunicaciones.

Se utiliza la información proporcionada por los sensores disponibles de tal forma que se pueda ir actualizando de forma errada o no, la posición del vehículo con respecto a las paredes encontradas y detectadas. El uso de los sensores podría utilizarse exactamente de la misma forma, es decir la plataforma móvil omnidireccional estaría en la capacidad de detectar, usando los algoritmos de esta aplicación y con un método SLAM, los caminos y posiciones que se realicen.

En [3] se desarrolla un sistema, en un entorno de viñedos y huertos, que utilizan robots autónomos en sus recorridos. Este tipo de entornos, en caminos reducidos, con falta de luz, y otros factores extremos, ha permitido que se doten a los vehículos (tractores autónomos) de varios tipos de sensores: GPS de frecuencia única, así como de sensores inerciales, cámaras y GPS diferencial, escáner laser, ruedas y codificadores de dirección, etc. Se utiliza un entorno maestro – esclavo, en donde el uso de dos tipos de agentes (un UAV y dos UGV's) es imprescindible. El UAV actúa como maestro y es el agente que más potencia tiene, al referirse al hardware y software que administra, en comparación con los UGV's que en tierra se desempeñan, de forma que el maestro entrega toda la información que tiene del mapa y la posición a los entes de tierra y ellos sin necesidad de procesarla (imágenes y caminos) la ejecutarían.

En este caso se ve ya la integración de varios tipos de robots y presenta también el resultado de una idea original que se presentó en el TFM realizado por el autor en el año 2012, que justamente involucraba la misma arquitectura en entono colaborativo con un dron y varios robots móviles en colaboración para tareas particulares.

Pero el concepto general de los robots multisensoriales y de múltiples propósitos se vuelve a repetir; robots que sepan “sentir” muchísimas cosas desde sí.

En la actualidad se ha demostrado la aplicación exitosa de sistemas que emplean la localización visual multisensorial [4] unida a un lazo de control realimentado dentro de una plataforma de procesamiento inteligente, que por su aplicación ha sido muy restringida, en donde los múltiples sensores actúan de forma redundante, lo que permite que se desarrolle un sistema de autonomía de baja potencia computacional. Su operación está basada en el uso particular y combinado de cuatro sensores especiales: Una cámara monocular, un láser LIDAR 3D, un sistema que combina sensores inerciales y un GPS y finalmente una cámara estéreo.

Dentro del procesamiento, por la imposibilidad de conocer previamente el camino y la línea de división del carril de circulación, se utiliza la cámara pasando su información por un bloque de filtrado estéreo o de orientación, que proporciona un procesamiento de imágenes por computador, de mejor resultado y prestaciones que las imágenes estáticas y una red neuronal convolucional; adicionalmente de forma simultánea se procesa la señal del LIDAR 3d obteniendo sus gradientes lineales. Los datos resultantes de los tres procesamientos son introducidos a un filtro de Kalman; el sistema no se basa en una sola señal medida sino en aplicar la redundancia de los sensores en una sola medición para obtener de forma definitiva certeza de la medición de la vía a seguir. Luego el procesamiento y la ejecución de los movimientos utilizan lazos de control realimentados.

Esto nos da la medida de poder desarrollar un sistema que utiliza simultáneamente las señales de la odometría proveniente de los encoders del robot y las señales del procesamiento visual de RGB de la cámara web, combinarlos en el algoritmo genético y encontrar las nuevas soluciones para llegar al resultado de eliminación casi por completo del error.

De forma minimalista y aplicada al campo de la educación, se han desarrollado robots, contruidos sobre la base de una plataforma educativa [5], con el fin de utilizar de forma eficiente los sensores que se pueden conectar a él; dentro de una plataforma de Makeblock (pequeñas plataformas, útiles y versátiles) que se controlan a través de una plataforma hardware simple basada en un microcontrolador tipo Arduino Mega.

A pesar de ser construido con plataformas comerciales de bajo costo, se le ha agregado capacidad de entregar objetos en un destino siguiendo una trayectoria determinada, tomado en

cuenta que pueda identificar la existencia o no de obstáculos en el trayecto; se ha añadido una función de reconocimiento de voz para que pueda seguir mediante ordenes una trayectoria, además de que pudiera encender las luces para cuando el ambiente este oscuro, en tal caso el robot dejaría de navegar y finalmente su control a través de un teléfono móvil.

Por la descripción anterior es lógico suponer que son utilizados múltiples sensores, orientados a las aplicaciones descritas, incluye: un sensor de luz alimentado a una entrada analógica con un valor de entre 0 y 1024; un sensor de sonido del medio ambiente que alimenta su entrada de igual forma a la placa de control mediante una señal análoga de la misma resolución anterior; luego utilizan también un sensor de efectos inerciales que es a la vez acelerómetro y giroscopio de tres ejes para la medición de la velocidad y posición del robot; un sensor de temperatura en circuito integrado para verificar que las partes y piezas del robot no se sobrecalienten, y un sensor de ultrasonido para encontrar los obstáculos con los que el robot se encuentre de manera frontal. Además de esto el robot cuenta finalmente con un sensor seguidor de líneas de lectura blanco/negro y encoders acoplados a las ruedas del robot con el objetivo de determinar su posición y sus movimientos.

En los experimentos en este pequeño robot se comprueba que las entradas de placas de bajas prestaciones inclusive, proporcionan información relevante para el control de las plataformas robóticas móviles, la diferencia con la arquitectura que se propone en esta tesis, tiene que ver más con los sistemas de control, también, inicialmente se había propuesto que los sistemas de control fueran obtenidos a partir de las placas de bajas, medianas o altas prestaciones computacionales y de hardware, pero por las necesidades de latencia, velocidad en las comunicaciones y sobre todo la necesidad de una elevada inteligencia que gestione los algoritmos de control que llevan en si una matemática y programación más profunda, hizo que el control sea desarrollado en una computadora remota, esta no necesita características muy elevadas, pero su naturaleza de procesamiento y comunicación le posibilita una ventaja sobre estos sistemas.

Muchas veces es necesario que se implementen sistemas en los cuales se realicen pruebas de medición de las variables de interés, y las respuestas que tengan estas a algoritmos en plataformas robóticas móviles, para ello se hacen necesarios construir sistemas de entrenamiento; así como en este trabajo de tesis se ha implementado una plataforma de entrenamiento (robot GYM), en otras aplicaciones y de forma más “completa” se han desarrollado estos sistemas; En [6] se describe la creación de un sistema base multitransportable, es decir que se pueda montar en cualquier tipo de robot móvil y sirva como elemento de entrenamiento del sistema para una aplicación particular; presenta el diseño, construcción y resultados de un banco de pruebas libres en el cual montar sus algoritmos y que se desarrolla

en cualquier tipo de plataforma móvil convencional; para ello está provisto de todos los elementos para su operación. Se habla de que el sistema es reconfigurable en operación y aplicación, de forma que puede colocarse en diferentes porta-vehículos y sensores. Es un sistema todo en uno que hace posible que toda prueba sea factible.

Utiliza un solo bus de comunicaciones, de tal modo que la comunicación sea simplista con cualquier plataforma conectada, posee también procesadores de cuarta generación que hacen posible incluso la programación desde la nube, y con la capacidad de procesar las señales de todos los sensores a bordo, incluido un SLAM de visión y una red neuronal de segmentación semántica.

El robot completo posee un sistema mecánico de liberación rápida, y la posibilidad de intercambio de plataforma, que mantiene un elemento amortiguador de movimiento y desplazamiento; en cuanto a lo electrónico posee un ordenador muy potente con un procesador i7 y un coprocesador o GPU NVIDIA GEFORCE 1070, con interfases de comunicación de 3 puertos USB 3.0, 3 puertos COM, 32 puertos DIO y 3 tarjetas SIM externas.

En cuanto a las modalidades sensoriales cuenta fundamentalmente con un LIDAR 3D, una cámara estéreo, una cámara de profundidad TOF, un receptor GNSS que permite localización satelital, un sistema de detección inercial IMU, un sistema de micrófonos de tipo arreglo, un odómetro de visión inercial que puede medir velocidad angular y aceleración en tres ejes; es decir un sistema completo para la detección de posición y movimiento del robot, además de ser sensores de muy altas prestaciones y de alto valor económico.

2.2 SEGUNDO EJE: ROBOTS MOVILES COLABORATIVOS

Una de las actividades más relevantes que se pretende en esta tesis, es el desarrollo de las aplicaciones de robots móviles en forma colaborativa, pues en ella se emplean medios y métodos especiales para la consecución de objetivos, En [7] se presentan robots que operan de forma remota y que están en ambientes peligrosos o lejanos para el ser humano, lo hacen a través de una interfase o una consola de operación, enviando y recibiendo señales que gobiernen los movimientos del robot. Lo que ellos denominan arquitectura maestro-esclavo que hace actuar al robot en el ambiente, conforme opera el humano desde la interfase (que en realidad se puede considerar una HMI); se ha considerado también en esta tesis en particular, pues se usa la misma arquitectura, con la diferencia de que no es el humano el que opera al robot sino un sistema inteligente que en muchos casos es una computadora remota en donde están residentes los algoritmos de control, y que de igual forma presenta los métodos de control para el robot desde

la estación, las comunicaciones, velocidad y retardos permitidos, así como la integralidad de los datos enviados y recibidos.

La ventaja de construir robots con tecnología completamente abierta es que permite controlar todas las características de movimiento, ya que cada sensor integrado, posee características especiales, en forma general, se hace que trabajen como sistemas redundantes que permitan la comprobación de la medición del parámetro deseado (movimiento, fuente de olor, trayectoria, etc.), esto a efecto de aumentar la precisión de los sistemas.

Las aplicaciones actualmente se extienden a robots de naturaleza diversa (aéreos, acuáticos, subacuáticos). En [8] se propone la existencia de tres tipos de robots: UAV que actuarían por encima del agua, a no muy alta distancia, USV que serían robots tipo barcos que flotarían sobre la superficie del agua y UUV, que sería robots submarinos que actuarían por debajo del agua.

Estos tres entes interactuarán de tal manera que el vehículo aéreo pueda transmitir las características del objeto sujeto de investigación que flota en la superficie (puede ser un iceberg, un naufragio, un bote con personas a bordo o un barco hostil), el vehículo que flota podría dirigirse en tareas de ayuda o abordaje del objeto inidentificado y el vehículo submarino, determinaría las características sub acuáticas que los demás no observan, como daños en la estructura, o si es que por debajo del agua tiene algunas formas o acarrea defectos o peligros potenciales al vehículo de interacción que flota sobre el agua.

La estructura determina que el vehículo subacuático tenga conectado sensores de profundidad como sonares e hidrófonos; el vehículo acuático sensores de cámaras y capacidad de procesamiento de imágenes y el dron, de igual manera posea cámaras de video o fotos más especializadas; de tal forma que el conjunto pueda trabajar en una red que monitoree y actúe sobre el objeto flotante bajo investigación. Las tareas del vehículo que actuaría sobre él son de variada naturaleza, desde guiado por un trayecto seguro al vehículo, rescate de naufragio o accidente aéreo y hasta el abordaje de personas desde el otro bote.

El objetivo es establecer la colaboración robótica entre múltiples dominios: sobre el agua que sería administrado por el UAV, en el agua, que sería administrado por el bote robot y bajo el agua que será administrado por el robot submarino. Al estar conectados los tres de una forma determinada para la colaboración, se deberá escoger los sistemas de comunicaciones determinísticos necesarios para poder establecer sin posibilidad de latencia o demasiados retardos las comunicaciones necesarias para la actuación eficiente; desde el robot submarino, un sonar de proyección de imágenes hacia arriba y mediante el hidrófono se inspeccionan las posiciones bajo el agua. La cámara montada sobre el UAV reúne la información fotogramétrica

de las posiciones sobre el agua y la cámara del bote, proporciona información de la forma y tamaño del objeto de enfrente y sus posibles colisiones.

Todos los robots se comunican entre ellos: en primer lugar, el robot bote con el robot submarino, a través de módems submarinos y entre el robot flotante y el dron a través de radiofrecuencias convencionales. En este caso la centralización de los datos la posee el robot bote y sobre él se desarrollarán las acciones de la colaboración, en tal sentido, el USV ayuda al UUV a localizarse, ya que al estar debajo del agua este no tiene acceso al GPS, al mismo tiempo que el UAV tiene un alcance de 200 pies sobre el agua y de esta manera puede ayudar a trazar un mapa al USV de varios kilómetros del entorno en donde se tiene que mover. El alcance de las comunicaciones entre uno y otro es el suficiente para mantener la colaboración en todo el tiempo.

Pero naturalmente las aplicaciones, a más de incorporar robots heterogéneos, también se centran en elementos de la misma especie, o que interactúan al menos en el mismo medio físico; [9] se centra en la utilización de pequeños robots voladores UAVs para el transporte de pequeñas cargas o telas ligeras con el fin de ayudar a personas a vestirse; esta es una aplicación muy relacionada ya que el trabajo de colaboración de los robots, permite determinar el cambio o no de la geometría del movimiento para llevar a cargo la tarea.

Vista la limitación que tienen los UAVs de la carga que pueden manejar, y la imposibilidad de introducir dentro de un vestuario sistemas muy grandes, se plantea la posibilidad cierta de usar enjambres de pequeñas UAVs para realizar esta tarea.

Es obvio suponer que este enjambre o alto conjunto de drones, deben realizar una tarea colaborativa a fin de utilizar su capacidad máxima de carga cada uno o al menos la tolerable, al mismo tiempo que ocupar un espacio determinado de tal forma que no exista la posibilidad de que haya colisiones entre todos los vehículos aéreos ni contra algún otro obstáculo que se le pueda presentar.

También se considera el hecho de que la tarea de colaboración final, será la de llevar la prenda de vestir o la tela hacia el destino indicado, evitando cualquier tipo de obstáculos que se presenten.

Dado que los tejidos suspendidos están fuertemente influenciados por el movimiento del aire, las fuerzas sobre los UAVs son en gran medida impredecibles y pueden ser de una magnitud bastante grande. En este caso, los UAVs que comparten la carga útil deben compensar cualquier perturbación en tiempo real. Esto puede lograrse haciendo que los UAVs cooperen y ajusten su movimiento de acuerdo con las posiciones y la dinámica (aceleración, par, etc.) de sus vecinos.

El sistema empleado emplea dos paradigmas fundamentales: la agregación de enjambres y las operaciones de vecindad, esto trata de agrupar a los robots en pequeños grupos de coincidencia de tarea, de tal manera que se unan grupos de robots que hacen tareas similares y que son opuestas por completo a otras tareas ejecutadas por otros robots.

Las operaciones de vecindad son tareas que realizan los grupos de robots o sub-enjambres con otros grupos de robots que hacen tareas distintas pero que pueden interactuar entre ellos para llevar a efecto una tarea mayor. En ambos casos lo neurálgico de la operación radica en las operaciones de comunicación entre los robots y el sistema, en la cual se define a más de la tarea a realizarse, la posición relativa que tiene cada uno de ellos en forma instantánea.

En una estación terrestre, tienen la información sobre todos los UAVs. Estas ubicaciones se alimentan en un algoritmo. El centro de comunicación emula la comunicación de los robots lo que da a cada programa su ubicación actual, y la ubicación relativa de sus vecinos. Mientras cada rutina en los UAVs en pasos continuos a través del script Buzz (algoritmo orientado al comportamiento de robots en enjambres), se gestiona la comunicación automáticamente, en la transmisión de todos los mensajes enviados entre cada UAV a través de su algoritmo.

Se especifica que para solucionar la problemática de la gestión de la máxima carga que tienen los robots de forma individual, y para que estos puedan, de forma colaborativa soportar su trabajo individual que genera el trabajo colectivo, se utiliza un control robusto de bajo nivel en el que se modela la carga máxima de cada UAV provocando un conjunto de interacciones locales que le llevan a un conjunto de interacción de toda la flota, de esta forma recuperando la robustez del sistema en conjunto.

El principio de operación se basa en que la carga útil puede ser aproximada por cables con peso uniformemente distribuido creando curvas de catenaria que cuelgan desde cada robot hasta un punto directamente debajo del centro de masa del enjambre, al que se refieren como punto de conexión. El vértice de cada curva de catenaria coincide con el punto de conexión.

Para que los UAVs mantengan una formación estable, usan el método de simular las fuerzas debidas a los resortes y amortiguadores que se encuentran entre cada robot. Usando el control con algoritmos de funciones potenciales artificiales que usan estas fuerzas de acción y reacción para la estabilidad del movimiento de los UAVs.

Habiendo tomado ya una referencia al uso de enjambres de robots, también en las aplicaciones se incluyen comportamientos de la naturaleza, para el control y coexistencia de múltiples robots, [10] se basa en el comportamiento de los seres de la naturaleza, para definir el comportamiento de un grupo de robots en cuanto a su forma de aprender la forma de moverse y evitar obstáculos a partir de un entorno desconocido de operación.

El principio fundamental de operación se denomina “tratamiento social de heridas” que se basa en el comportamiento natural de las hormigas de cierta especie, en su batalla contra las termitas. Se considera que las hormigas heridas en la batalla y que no están muy malheridas, pueden ser rescatadas por sus compañeras llevándolas al nido para curarlas y que se recuperen. El sistema comprueba que la tasa de mortalidad de las hormigas rescatadas es del 10% sobre el 85% de mortalidad de las que se dejan en el campo de batalla abandonadas.

Para el sistema de navegación y evasión de obstáculos en un grupo de robots, se usa un sistema descentralizado con N robots homogéneos que se comunican entre sí. El objetivo de este sistema descentralizado es encontrar los parámetros óptimos, que parametrizan el controlador de cada robot para aprender la tarea de evitar obstáculos y moverse libremente por el entorno. Para simplificar, los robots están equipados con los sensores y actuadores necesarios para realizar mediciones básicas de distancia, así como traslaciones y rotaciones en el entorno en el que actúan.

Es recurrente que las ideas de muchos experimentos propuestos en el transcurso del desarrollo de esta tesis, hayan sido desarrollados como experimentos en otras instancias y se hayan publicado con mucha aceptación.

Cuando se habla de robots móviles, se tiene que determinar también sus características, que sean una ventaja al momento de emplearlas en aplicaciones de diversa índole, específicamente en la navegación y tareas relacionadas; [11] divide una región de interés completa en varias áreas o secciones en las que se pone a actuar a un robot móvil no holonómico, y según la potencia o debilidad computacional y de hardware que este posea podrá identificar el entorno en donde se mueve, asignándoles a los de menor potencia sectores de actuación más pequeños.

Los agentes aprenden sus propios parámetros de desempeño mediante el uso de un estimador; para ello la información será tratada por una red neuronal Hopfield, por su alto nivel de precisión.

El sistema está basado en la ley de control que se define por el movimiento hacia el centroide, la colisión entre robots puede producirse por el uso de radios de movimiento más grandes de los determinados, incluidas las dimensiones del robot. Los parámetros de estimación en línea se obtendrán de la derivada matemática del estimador.

En las pruebas de simulación se comprueba que, mediante el algoritmo, se asignan regiones más pequeñas a agentes de menor capacidad, o degradación en su rendimiento, y en los experimentos se comprueba a través de las áreas de cobertura, que los agentes que tienen mayores recursos asignados mediante un parámetro al algoritmo, se les asigna una mayor área de cobertura que los que no.

Se hace mayor énfasis en los algoritmos y los procedimientos utilizados como método de colaboración entre robots, más que hardware que se utiliza en ellos, es así que casi ni se lo menciona. Pero definitivamente, en forma general, se hace alusión a ambos procesos ya que un diseño efectivo de los robots hace que los algoritmos sean probados de forma más eficiente y de la misma manera la forma operativa eficiente (lograda mediante hardware) hace que los resultados sean más visibles al momento de presentarlos.

Finalmente se tiene que analizar también que la aplicación de redes de robots, y las tareas colaborativas, incluyen robots de tipos heterogéneos (principal idea que este trabajo de tesis desarrolla) [12] se basa en el uso de una red de múltiples robots móviles heterogéneos, que tiene que desarrollar un juego de tareas asignadas a todo el conjunto. La red deja de operar por si misma cuando todas las tareas han sido realizadas, aunque algunas de ellas hayan sido realizadas por más de un robot a la vez.

Se considera a cada uno de los robots de la red como un nodo de red inalámbrica, que efectuada su tarea o por características intrínsecas de él puede desconectarse de ella (fallas, o pérdida de señal en el área de trabajo). Existen varios tipos de robot en esta red, diseñados para realizar diferentes funciones, según su naturaleza (hardware o software especializados u orientados); esta característica diferente proporciona una ganancia específica de cada robot en el modelo matemático de la red para la asignación de su tarea; si en el proceso de asignación de tareas dos robots candidatos tienen las mismas capacidades y ganancias, se determina su nivel de carga de energía para realizar la asignación y se modela la que necesitaría para llegar y ejecutar la tarea.

2.3 TERCER EJE: REDUCCION DEL CONSUMO DE ENERGIA EN ROBOTS MOVILES

El consumo y ahorro de energía en robots móviles es un tema altamente discutido, no solamente por los afanes de desarrollo comercial o económico, sino más por la determinante característica que representa este parámetro en el trabajo de este tipo de robots, al mismo tiempo es un tema que ha dado muy pocos resultados nuevos también, ya que, en cuanto a hardware, por ejemplo, muy poco se ha podido hacer, a no ser la investigación de fuentes más durables y mejorar las tecnologías para la conservación de la energía. [13] Se basa en la utilización de redes de sensores y actuadores que consumen su propia energía y necesitan recargarse; En las WSN, para recargar los actuadores, una estrategia sencilla es dejar que se trasladen a la estación de suministro de energía para su reposición. Sin embargo, no es factible dejar que los actuadores dejen de ejecutar las tareas y se desplacen fuera del área de cobertura. Esto se debe a que esta estrategia no sólo hace que los actuadores no puedan responder rápidamente a los eventos

detectados, sino que también daña la red entre actuadores. Otra forma de recargar los nodos de los actuadores es darles mayor prioridad que a los sensores ordinarios. Sin embargo, esto puede resultar en muchos nodos de sensores no funcionales y, por lo tanto, en la desconexión de la red debido al largo tiempo necesario para recargar los actuadores a la capacidad total de su batería. De hecho, el diseño de un método de recarga razonable para las WSN es un gran reto, ya que hay que tener en cuenta los siguientes factores: En primer lugar, los consumos de energía de los sensores y actuadores muestran una propiedad altamente dinámica porque interactúan íntimamente con el entorno. Dado que las soluciones de carga periódica de offline para el cargador móvil pueden tener un rendimiento de carga deficiente en entornos de red reales, resulta difícil recargar los nodos (incluidos los actuadores y los nodos de los sensores) para cumplir con el plazo de entrega de la batería de los nodos.

En segundo lugar, debe tenerse en cuenta la significación de los actuadores para las WSN. Un método de carga razonable debería asegurar el suplemento de energía para todos los actuadores solicitados. Tercero, la equidad es un tema importante y desafiante en el proceso de carga. Además de asegurar que no haya fallas en el actuador, los sensores deben recargarse oportunamente con el fin de evitar la mayor cantidad posible de fallas en los sensores. En cuarto lugar, además de recargar los nodos de forma oportuna y eficiente, el coste de recarga del cargador móvil, debería reducirse al máximo. No existe relación general con nuestro trabajo.

Por otro lado, los robots móviles también deben desarrollar tareas específicas, y no solamente de comunicaciones, sino también tareas “manuales” y altamente operativas, que son de hecho las que consumen mayor cantidad de su energía (levantado de objetos, transporte de objetos pesados, etc.) [14] considera el almacenamiento de objetos por parte de un robot móvil, que se encuentra entre la estantería y un pasillo transversal, en dependencia de la carga de su batería, si está descargado solamente ejecuta su trabajo debajo de la estantería. La estrategia que produce la reducción en el consumo de energía o mejor dicho la maximización de la energía del robot, en su operación, se basa en: el almacenamiento aleatorio, según su clase y según su posición de almacenado. He aquí la importancia de una estrategia operativa, sobre la persistencia operacional del hardware; en ella, los robots utilizados contienen seis ruedas donde las cuatro esquineras que son omnidireccionales, no tienen potencia sino solamente estabilidad y navegación; solamente dos ruedas colocadas en el centro contienen la tracción del robot. Cuando el robot se mueve de forma directa, estas ruedas le dan la direccionalidad y se mueven a velocidad constante, pero cuando el robot realiza una curva, estas ruedas tienen una velocidad diferente.

Cuando el robot se mueve de forma recta tiene más fuerza de tracción en las ruedas interiores y las cuatro ruedas omnidireccionales solo proporcionan balance en este caso la elevación y

transporte de la carga es liviana y no usa mucha energía; el consumo de energía está solamente en dependencia del número de lotes de carga que el robot maneje, en este sentido, no se establece una estrategia que disminuya por sí mismo y de forma inteligente el consumo de energía, sino un sistema que asigna a varios robots un número determinado de carga para que el manejo de esta sea la que discrimine el uso de energía de cada robot.

Entonces se hace más bien un manejo de la eficiencia de la energía entregada a todo el sistema en su totalidad y se define que la asignación de tareas en menos proporciones a robots cargados o descargados hace mejor y más eficiente el uso de la energía.

En aplicaciones más novedosas y actuales, se emplean nuevos mecanismos que procuren el deseado ahorro de energías, en [15] se considera la reducción en la cantidad de energía que consumen dispositivos móviles, realizando la comunicación entre ellos por la técnica de retrodispersión, para vehículos terrestres no tripulados. Se explica que, con el avance paulatino y determinante del Internet de las cosas, cada vez más elementos colgados en la red necesitan de consumo de energía; mediante la eliminación de la RF por la tecnología de retrodispersión se mejora el aspecto de rendimiento, a la vez que al eliminar el trayecto de ida y vuelta el rango de transmisión se ve profundamente comprometido. La solución planteada crea un balance entre la energía que se consume por el movimiento del robot y la energía que se consume en las transmisiones de los datos entre elementos.

En cambio, al abordar completamente el hardware nos encontramos también con soluciones actuales y más versátiles, dependientes también de la tecnología actual. [16] propone la mejor utilización de energía en un vehículo eléctrico, mediante la asignación de un control óptimo de energía a las ruedas, utilizando un seguimiento óptimo de caminos; usa una técnica denominada vectorización del torque que minimiza la resistencia al movimiento del vehículo especialmente en las curvas, con las lógicas consideraciones de consumo de energía de las ruedas. Tiene como principal objetivo el mejorar el estado de carga de la batería y ampliar el alcance del movimiento del vehículo, el método es conocido como LPV uso de parámetros lineales variables.

Aquí se relaciona directamente el control de seguimiento de trayectoria de un vehículo eléctrico con su uso óptimo de energía, basado principalmente en un sistema reconfigurable SOC que maximiza el uso de la energía de la batería, con el fin de lograr el máximo alcance del vehículo, usan un método llamado seguimiento de carriles reconfigurables e implementan para ello un controlador, aquí la asignación del control de alto nivel de la dirección y la generación del momento de giro tiene como objetivo minimizar la resistencia de las curvas; adicionalmente un controlador de bajo nivel que optimiza de forma “multiobjetivo” el par motor de cada rueda haciendo que el consumo de energía sea mínimo, teniendo en cuenta la eficiencia del motor eléctrico y la certeza del movimiento..

En [17] el sistema de control del movimiento e incertidumbre está basado en la teoría de estabilidad de Lyapunov y el uso de algoritmos de control inteligente basados en el aprendizaje en línea, específicamente las técnicas de lógica difusa con aplicación de la transformada wavelet en redes neuronales. Además de todo aquello se introduce un modo de control de deslizamiento para reducir las perturbaciones y adaptar el controlador a problemas de seguimiento de caminos del robot.

El algoritmo HS (harmony search) reduce la probabilidad de que exista una sola solución, sino un vector que se acerque más a ella con el uso del algoritmo genético, este asegura la solución global óptima a partir de ese conjunto. Este opera inicializando y calculando los pesos que se asignará a los valores individuales, que luego serán ordenados en ponderación y elegidos de forma que se acerquen a la solución; luego de ello, se los hace evolucionar con el procedimiento de cruce y se determina la tendencia que ellas tienen a la solución; de esta forma en las nuevas generaciones se ve la solución.

Finalmente se han realizado trabajos también en el orden de usar la robótica colaborativa en el desarrollo de una estrategia que guíe a la reducción del consumo de energía de enjambres de robots. En [18] se introduce una infinidad de robots móviles en trabajo colaborativo, habla de enjambres de robots móviles encargados de limpiar el medio ambiente, explica que los robots recogen los escombros encontrados en la arena de la playa, la energía de los robots, por supuesto disminuye el momento de ejecutar su tarea particular, y de moverse hacia ella; es necesario que un sistema determine que la energía de trabajo del robot individual, no llegue a un nivel más bajo que un umbral límite, pues en este caso, al no poder ya realizar las tareas, de movimiento primordialmente y limpieza en segundo lugar, se pone en modo de carga.

Cuando esto ocurre el robot se traslada a un área de carga, en donde se recarga de energía por sí mismo, teniendo la posibilidad de compartirla con algunos otros robots que tengan menor cantidad de energía; la forma de compartir energía entre robots se basa en el modelo trofaláctico, usado en las colonias de termitas para compartir su alimento.

Usa técnicas de algoritmos genéticos para obtener un conjunto particular de valores que sean solución que determine el menor consumo de energía, manifiestan que este modelo mejora el rendimiento del enjambre en términos del consumo de energía.

La robótica de enjambres, es una técnica muy utilizada que experimenta con los modelos de comportamiento de grandes conjuntos de animales para desarrollar tareas específicas, en este contexto la aplicación de un gran conjunto de robots limpiadores de playas, resulta ideal para ser aplicada con esta técnica; en ella, no existe un líder que guíe los pasos de todo el conjunto de

animales, sino más bien todo el enjambre actúa de una forma conjunta, como que tuviera un solo pensamiento.

La operación de los robots de forma individual, se resume en la siguiente: Inicialmente el robot cargado se encuentra en el estado de búsqueda hasta encontrar una basura en su entorno, una vez que la encuentra cierra su pinza, recogiendo la basura y se traslada a la zona de vertedero donde vuelve a abrir su pinza, depositando la basura en su lugar; una vez hecho esto vuelve hacia su estado de búsqueda, repitiendo continuamente esta tarea mientras su nivel de carga no esté más abajo del crítico; si esto sucede se traslada a la zona de carga, al llegar a ella se carga por sí mismo y vuelve a repetir su procedimiento. Si al regresar encuentra un robot que tiene suficiente carga para compartirla, intercambian la carga y regresa a su operación de búsqueda normal.

Para reducción del consumo de energía el sistema evalúa cuando los agentes participan en la acción trofaláctica que es la que conduce a la minimización del consumo de energía, pues evita el movimiento completo hacia la zona de carga y su consiguiente pérdida de energía.

2.4 CUARTO EJE: CORRECCION DE ERRORES DE ODOMETRIA PARA ROBOTS MOVILES.

De forma similar a lo tratado en los tres ejes de actuación anteriores, la localización precisa de un robot en su entorno de trabajo, proporciona la seguridad en la tarea posterior que va a realizar, sea de forma individual o colectiva, en este sentido se han desarrollado trabajos relacionados a la reducción de errores, de localización de robots móviles, tema que ha sido considerado muchísimas veces en todos estos años y que es tema de debate aun hoy en día. [19] se centra en la fusión de datos de odometría; al hablar de fusión de datos hace referencia al uso simultáneo de información proveniente de varios sensores, la odometría es el estudio de la estimación de la posición relativa a su localización inicial en vehículos con ruedas. En una primera instancia explica acerca de la compensación de fallas del sensor en la medición mediante odometría, misma que genera un problema de deslizamiento sin embargo el sensor corrige esta falencia para luego procesar la información mediante la ayuda del Filtro UKF. La segunda etapa se implementa el algoritmo AMCL, pueden utilizar el mapa y el láser conocido de radar para realizar una segunda corrección de postura en el robot, pese a ello queda un pequeño error que da inicio al uso del algoritmo END. La tercera etapa se la realiza mediante escaneo láser y lo convierte en el formato END para que se adapte a la segunda etapa logrando así un sistema de corrección de posición de tres capas.

El modelo físico del robot a ser controlado incide directamente ya que se necesita los parámetros tales como el radio de la rueda robot y la distancia entre ejes. Para evitar que el error acumulativo aumente con el tiempo se debe implementar sensores externos.

El robot está equipado con una odometría, IMU y un radar láser 2D, además consta dentro de él, un IPC (Industrial Personal Computer), junto con una cámara infrarroja exterior que proporciona el acceso visual para aplicar la navegación autónoma. El ambiente experimental que ha sido seleccionado para las diferentes pruebas es al aire libre con un gran número de desiguales: césped, tierra, propios del medio ambiente que hacen el entorno más complicado; de esta forma se logra la obtención de datos de manera más real con respecto a resultados esperados.

El modelo híbrido que utiliza el procedimiento para la corrección de errores de odometría es importante para el trabajo de tesis que se presenta, pues muestra una alternativa similar, a pesar de que en este sistema usan dos niveles de redundancia, también es factible y más económico realizar un sistema híbrido que usa otra alternativa de medición para la corrección.

El trabajo desarrollado en [20] expone algunos métodos y algoritmos, uno de ellos es la calibración que es capaz de resolver el problema de la perspectiva simplificada desde tres puntos y la perspectiva de tres líneas, respectivamente. Otro método es la localización de vehículos en el que se pueda fusionar datos de múltiples sensores como un sistema estereoscópico, un telémetro láser y GPS, etc. Los autores proponen un nuevo algoritmo que se basa en la agrupación de pirámides espaciales para segmentar objetos de manera robusta a múltiples escalas y mejorar la localización de los límites de los objetos mediante la combinación de métodos de redes neuronales convolucionales profundas y modelos gráficos probabilísticos para combinar las respuestas en la capa final de DCNN con un campo aleatorio condicional completamente conectado (CRF).

Cada uno de estos algoritmos propuestos se prueba en diferentes conjuntos de datos y se realizan experimentos reales en la plataforma del robot móvil el cual fue desarrollado internamente y equipado con múltiples láseres, PC industriales, GPS, cámara monocular y sistema de navegación inercial, el que permitirá demostrar que la imagen RGB-DI y el enfoque basado en el modelo FCN pueden proporcionar un rendimiento superior para la comprensión de la escena al aire libre. El modelo FCN propuesto contiene tres capas deconvolucionales, cada una de las cuales está apilada con la información de la capa intermedia. Donde la convolución y la deconvolución del modelo FCN realizan la codificación y decodificación de la entrada de la imagen RGB-DI.

Las redes neuronales convolucionales profundas (CNN) están en el centro de muchos algoritmos de clasificación y segmentación de vanguardia en visión por computadora. Las técnicas basadas en CNN logran precisiones previamente inalcanzables por métodos clásicos. Sin embargo, en la robótica móvil y la estimación de estado, no está claro hasta qué punto estas arquitecturas profundas pueden obviar la necesidad del modelado geométrico clásico.

En [21] se combina la odometría visual (VO) y la tarea de calcular la egomoción (movimiento de la cámara en imagen rígida) de una cámara a través de un entorno desconocido sin fuentes externas de posicionamiento. Los algoritmos de localización visual como VO pueden sufrir varias fuentes de error sistemáticas que incluyen sesgos del estimador, calibración deficiente y factores ambientales.

Demuestran que las CNN pueden inferir cantidades geométricas difíciles de modelar (por ejemplo, la dirección del sol) para mejorar una estimación de localización existente. Por lo tanto, propusieron un sistema que toma como punto de partida un algoritmo de localización eficiente y clásico que calcula estimaciones de pose de alta velocidad. A esto, agregaron una Red de corrección de postura profunda (DPC-Net) que aprende correcciones "bajas" de baja tasa de los datos de entrenamiento que luego fusionaron con las estimaciones originales; la arquitectura DPC-Net se puede modificar fácilmente para aprender las correcciones SE (3) (Special Euclidean Group); que representa las posiciones del robot, obtenidas de la matriz de transformación; para estimadores que operan con otras modalidades de sensores (por ejemplo, lidar). Para esta tarea general, derivaron una función de pérdida de regresión de pose y una expresión analítica de forma cerrada para su jacobiano. Esta pérdida permite que una red aprenda un vector de coordenadas de álgebra de Lie sin restricciones, pero deriva su jacobiano con respecto a la distancia geodésica SE (3).

DPC-Net, es una forma novedosa de fusionar el poder de redes profundas y convolucionales con procedimientos de localización geométrica clásica. Utilizando una nueva función de pérdida basada en grupos de matrices Lie, DPC-Net aprende SE (3) y corrige las posiciones, para mejorar un estimador de referencia y mitiga el efecto del sesgo del estimador, los factores ambientales y las malas calibraciones del sensor. Además, puede hacer que un procedimiento de odometría visual estéreo dispersa sea tan preciso como un estimador denso de última generación, y mejorar significativamente las estimaciones calculadas con un modelo de distorsión de lente mal calibrado.

Por otro lado, en un entorno desconocido, la evaluación de la trayectoria del robot en tiempo real es una de las claves para el éxito de una misión robótica. En un entorno de este tipo, las mediciones absolutas, como los datos GPS, pueden no estar disponibles. Además, la estimación de la posición utilizando únicamente sensores propioceptivos, como los encoders y las unidades

de medida inerciales (IMU), generará errores que aumentarán con el tiempo. [22] presenta un enfoque de fusión multisensorial entre IMU y el flujo óptico en tierra utilizado para estimar la posición de un robot móvil, al tiempo que se asegura una localización de alta integridad. La fusión de datos se realiza a través de la forma informativa del filtro Kalman, es decir, el filtro de información (IF). Se añade un paso de detección y exclusión de fallos (FDE) para excluir las mediciones erróneas del procedimiento de fusión, haciéndolo tolerante a fallos y garantizando un alto rendimiento de localización. El enfoque se basa en el uso del Marco Integrado para la estimación del estado y en las herramientas de la teoría de la información para el FDE. El enfoque evalúa la calidad de una medición basada en la cantidad de información que proporciona utilizando métricas informativas como la divergencia Kullback-Leibler.

Para navegación cerca de seres humanos se necesitan estimaciones de posicionamiento precisas para una operación exitosa; sin embargo, al fusionar mediciones de múltiples fuentes, pueden surgir algunos problemas, como imperfecciones de los datos y errores de medición. Estos problemas pueden dar lugar a una estimación inexacta que implique la necesidad de excluir los datos erróneos del procedimiento de fusión.

Las técnicas de odometría visual basadas en el seguimiento del movimiento entre dos imágenes han demostrado su eficacia para localizar robots durante operaciones planetarias, misiones submarinas o navegación en carretera. Los nuevos enfoques en odometría visual han utilizado diferentes métodos para ser robustos a varios y pobres niveles de luz, tales como técnicas basadas en eventos o descriptores binarios. Sin embargo, las soluciones basadas en el uso de una cámara estándar tienen desventajas, tales como el alto costo computacional del procesamiento de imágenes, su costo para aplicaciones robóticas y su incapacidad para hacer frente a un amplio rango dinámico de condiciones de luz.

Como consecuencia, la odometría visual minimalista basada en un tipo de sensor único se desplazará drásticamente. Para hacer frente a la restricción de cada tipo de sensor, en este trabajo, proponen fusionar la FP con las medidas IMU.

Estos métodos que se encuentran en la literatura requieren un conocimiento a priori de los modelos de falla, información sobre los sensores en su operación normal, expertica, o la disponibilidad de datos suficientes.

Las principales contribuciones de este documento son las siguientes:

- Fusión de datos entre las mediciones IMU y OF para asegurar una localización de alta integridad de un robot móvil;
- El uso del FI para la fusión de datos y del KLD para el FDE;

- Estudio experimental del método desarrollado en un entorno exterior bajo diversas condiciones, incluyendo iluminación de alto rango dinámico y diferentes texturas de suelo;
- Comparación con resultados anteriores basados en el KF y el filtro de mediana.

La principal ventaja de este procedimiento de información es que será capaz de detectar múltiples fallos a la vez con un tiempo de procesamiento ventajoso gracias al modelado del paso de actualización del EIF como una simple suma de las contribuciones de información de diferentes observaciones. Además, el residuo basado en el KLD tiene en cuenta diferentes tipos de información (divergencias de matriz de Mahalanobis y Burg) entre las distribuciones previstas y las de actualización.

Finalmente, las técnicas de SLAM son ampliamente utilizadas y difundidas para robots en ambientes desconocidos, como [23] que realiza un novedoso algoritmo de localización y mapeo simultáneo visual inercial y monocular estrechamente acoplado, que proporciona un seguimiento preciso y robusto del movimiento a altas velocidades de fotogramas en una CPU estándar.

La estimación de movimiento concurrente y la reconstrucción de mapas mediante la combinación de mediciones visuales e inerciales ha recibido un interés significativo en el campo de las comunidades de robótica y visión por computador. Este conjunto de sensores visuales-inerciales puede servir como una alternativa ideal al GPS en entornos donde se niega el GPS, ya que ambos sensores son pequeños, ligeros, baratos y complementarios.

Por un lado, la localización y el mapeo visual simultáneo (SLAM) pueden proporcionar un buen seguimiento del movimiento y una rica información cartográfica en entornos visualmente distinguibles. Sin embargo, debido a la limitación del sensor, el SLAM visual es sensible al movimiento borroso, las oclusiones y los cambios de iluminación. Por otro lado, los sensores inerciales pueden proporcionar información de movimiento propio a alta frecuencia, por lo que la navegación inercial es robusta a los movimientos agresivos y es capaz de proporcionar una escala absoluta para el movimiento. Mientras que el resultado de la navegación inercial es ruidoso y diverge incluso en pocos segundos.

Esto nos hace observar que fusionar estrechamente las mediciones desde el sensor inercial hasta el SLAM visual se mejorará la robustez y la precisión de forma drástica. Esta ventaja se acentúa más cuando el sistema es monocular, pues la escala del movimiento estimado y el mapa construido es ambiguo y se puede desviar con el tiempo.

Este sistema extrae únicamente características de fotogramas clave para manejar todo el conjunto a alta frecuencia, pero muchas veces esto no es suficiente, por lo que se extrae más

características de los fotogramas, lo que resulta en más correspondencias de características de información, que aumenta la precisión.

Para concluir el análisis vemos que [24] utiliza una red neuronal profunda y un sistema de medición y procesamiento visual RGB-D que hace que se tenga una ventaja en la información de la profundidad en la información visual, además que usa también elementos inerciales para la medición de los desplazamientos de los robots.

Una red llamada VIO Learner es entrenada en base a parámetros unitarios de VIO sin usar los elementos inerciales de desviación del giroscopio y ruido del acelerómetro; esta red plantea la solución parcial de las medidas del desplazamiento, que luego son corregidas por los jacobianos de los errores de proyección de las imágenes a escala que se van produciendo.

Los enfoques en VIO combinan estimaciones visuales del movimiento con los medidos por acelerómetros multieje y giróscopos en unidades de medida inerciales (IMU). Como IMUs miden sólo las aceleraciones lineales y las velocidades angulares, los enfoques inerciales de la localización son propensos a ser exponenciales la deriva en el tiempo debido a la doble integración de las aceleraciones en estimaciones de pose.

Para realizar este trabajo utilizan un cambio de posición entre imágenes con una arquitectura jerárquica y estimación a escalas múltiples, que luego se corrigen utilizando un procesamiento convolucional del jacobiano de la proyección euclidiana del error de desplazamiento con respecto a las coordenadas del origen. Además, que al utilizar RGB-D se incluye la profundidad de las imágenes para lograr la recuperación absoluta de la escala, mientras que la profundidad absoluta puede lograrse incorporando un sensor a bordo. Su principal contribución es el entrenamiento y aprendizaje no supervisado de las estimaciones de trayectoria con recuperación absoluta de la escala a partir de mediciones RGB-D + inerciales con:

- Módulos de corrección de errores en línea incorporados;
- Extrínsecos desconocidos de la cámara IMU; y
- Cámara y IMU de sincronización temporal flexible.

Para no entrar en más detalles técnicos se observa que este método es más profundo y que utiliza ya una cámara especializada para mirar los detalles de las imágenes y reconstruirlas, pero también hace un modelo híbrido entre la información/procesamiento de la cámara RGB-D y los elementos inerciales montados a bordo del robot.

Capítulo III

3 DISEÑO Y CONSTRUCCIÓN DE LAS PLATAFORMAS ROBÓTICAS MÓVILES MULTISENSORIALES.

Como se ha mencionado ya en las páginas anteriores, la experiencia en el desarrollo de robots móviles autónomos en tareas de búsqueda, realizada como trabajo de fin de máster, dejó como resultado las interrogantes de construir plataformas más robustas, que superen las limitaciones del robot que fue construido en esa etapa (Lucy01), y que presentaba problemas de deslizamiento en las ruedas motrices tanto al arranque como en superficies lisas e irregulares, además de que las plataformas embebidas usadas ya tenían al momento, sus versiones mejoradas y más aplicaciones dentro de sus características, especialmente del sistema operativo y la interconectividad; no se había evaluado tampoco la eficiencia de un sistema de comunicaciones que posibilite la existencia de más de una plataforma y peor aún de pensar en que la búsqueda sea multiagente y en entorno de colaboración. Otra de las limitaciones era que el paso de los robots de ambientes indoor a outdoor no había sido considerado y menos aún la posibilidad de trabajar en ambientes parcial o totalmente no estructurados y desconocidos.

Todo esto plantea la necesidad de construir nuevas plataformas que permitan solucionar todas estas limitaciones y comprobarlas prácticamente con el desarrollo de varias aplicaciones inteligentes y muy complejas que pongan a prueba y al límite de operación estos nuevos robots y se sepa con certeza que se resolvieron todos sus problemas, más que únicamente demostrar que si se pueden construir y además pueden operar, porque siendo solamente este el concepto, para nada serviría la mejora. Es así que lo primero que se atacó es el hardware en toda su dimensión: es decir la electricidad y la electrónica de movimiento y control necesaria para la operación de los robots con características particulares específicas para cada una de estas aplicaciones diseñadas, concomitante con esta circunstancia y no menos importante, hacía falta también un diseño que, en lo mecánico solucione problemas que se presentaban en los robots, esencialmente al momento de introducir errores especialmente en su localización, ya que en la mayoría de los casos, los errores de odometría son introducidos en un sistema robótico autónomo por la falla del diseño, como longitudes de los ejes, etc. (se dedicará en una tesis un acápite especial a las soluciones encontradas).

En un inicio, para el control autónomo del robot Lucy01 se utilizó una placa embebida Beaglebone [25] original, que se eligió por sus muchas prestaciones, se la escogió de entre otras alternativas como IGEPV2 y raspberry [26]. Para el inicio del proyecto de investigación de doctorado, muchas más plataformas de este tipo y naturaleza aparecieron, evoluciones importantes de Beaglebone, Raspberry Pi, Banana Pi, IGEP v2, Flora, Lilypad, UDOO, y

muchísimas otras más, conformando un amplio espectro de tarjetas de desarrollo; sin dejar atrás las FPGA's y sus módulos de desarrollo también.

En este contexto había mucho trabajo, con el elegir, cuál sistema presentaba las mejores prestaciones y si algunos de ellos de alguna forma tenían similares características; había que someterlos a pruebas de operación para determinar cual tenía mayor flexibilidad, capacidad computacional y menos consumo de energía.

Es así que se empleó un criterio normalizado, de uso de placas de desarrollo de iguales prestaciones (las más necesarias y con características de vanguardia), para poner a prueba nuevos elementos, que proporcionen ventajas en las nuevas plataformas a desarrollar y que al mismo tiempo contribuyan ya a conseguir los objetivos deseados en el proyecto.

Se analizan las características de los elementos mencionados, y se eligen tres plataformas de desarrollo de similares características, con el fin de montar en ellas todos los algoritmos que se necesiten para mover y controlar el robot móvil, con diferentes alternativas. Esta elección se realiza tomando en cuenta los siguientes parámetros:

- Posea internamente un procesador con tecnología RISC de preferencia un ARM Cortex de nueva generación.
- Se pueda descargar o establecer un sistema operativo, que actúe como un RTOS dentro del sistema embebido.
- Que posea múltiples I/O digitales y algunas (mínimo 8) I/O analógicas.
- Que posea un set de comunicaciones electrónicas basadas en I2C, SERIAL, USB, Wireless LAN, SPI, etc.
- Lectura del sistema/almacenamiento microSD.
- Alimentación de baja potencia de 3.3Vdc.
- Leds indicadores de estado y operación claros y definidos.
- Memoria mínima de 512 MB.
- Ethernet 10/100.
- Capacidad de operación de LCD.
- Puertos seriales, timers y comunicaciones por redes especiales de preferencia CAN bus.
- Peso y tamaño pequeños, y
- Amplio conjunto de sockets de conexión de GPIO.

Tomando todo esto en consideración, se decide utilizar en los primeros protocolos de prueba las tarjetas: Beaglebone Black serie C; Raspberry PI B+, Banana PI M4 y IGEP v2.

Con la finalidad de hacer trabajo colaborativo con los robots, interesaba saber el consumo de energía de las baterías del arreglo total y de cada uno de ellos, para saber a ciencia cierta su

consumo en las tareas que tendría que ejecutar. Para ello se decide hacer un estudio de las clases de baterías existentes, nuevas tecnologías de baterías y medidores de carga.

Se realizó de manera parcial, un primer estudio, en donde se determinó que las baterías más eficientes en esos momentos, eran las baterías de Litio (iones de litio) pero se no se profundizó en esto porque se pretendía primero saber realmente cuanta batería podía consumir un robot estándar para cada tarea de movimiento y búsqueda (inicialmente, como problema ya resuelto), y probar dicho rendimiento en diversas plataformas de control, de esta forma tener una visión general del consumo de cada robot con cada plataforma, o de un mismo robot (sistema móvil) con cada plataforma de control.

El problema del consumo de energía de los robots móviles, sigue aún sin resolverse en el mundo, han habido estudios recientes incluso, de sistemas inteligentes que integran algoritmos genéticos, que se han desarrollado para “asegurar” que un robot nunca pierda su carga en los trabajos que realiza; esto quiere decir entonces que es prácticamente irrelevante un estudio de los mejores sistemas de alimentación eléctrica que permitan hacer sustentable el trabajo de un robot, especialmente de uno móvil, por lo que se han hecho investigaciones de las baterías más eficaces al momento de usar este tipo de autómatas, y a partir de ahí desarrollar técnicas que permitan el ahorro de energía, es decir sea alimentado con un elemento de pésima duración hasta con uno de larga duración. En otras palabras, no importa verdaderamente cuanta carga pueda proporcionar una batería, frente a una buena estrategia de hacer durar eficientemente dicha carga;

En resumen, esta es la razón, y después de haber realizado ensayos con algunos tipos de baterías en aplicaciones iguales, a partir de ese momento se decidió usar baterías LIPO (polímero de litio) en todos los robots, a pesar de ser baterías delicadas y mantener cuidados especiales para evitar accidentes, especialmente incendios; son baterías de gran rendimiento en cuanto a capacidad y tensión.

Por otro lado y resuelto el problema de las baterías, la finalidad es incorporar a un robot, la mayor cantidad posible de sensores de diversa índole y aplicaciones de tal manera que se asemeje a un ser humano, que cuando carece de un tipo de sentido, este es suplido por otro inmediatamente, P.E. a falta de luz, la localización de una persona puede hacerse mediante el tacto, el oído o el olfato; especialmente cuando los robots trabajan en ambientes desconocidos, es decir incorporar también a ellos características de SLAM. Afortunadamente en la actualidad, los fabricantes de todo tipo de sensores los construyen, incorporando a ellos un estándar de comunicaciones, es decir se pueden conectar directamente a puertos estándar como RS232, USB, etc. O hablar protocolos de red e interfases de conexión masiva, como SPI, I2C, etc. En este contexto se debe asegurar que el elemento de control posea todos estos recursos, para poder: en

primer lugar, conectarles a todos los sensores proporcionados, y luego que operen e incluso que se intercambien sin dificultad, y así realizar la medición simultánea de las señales de estos sensores; si analizamos las opciones elegidas, todas ellas poseen los requisitos antes mencionados.

En el afán de determinar los sistemas óptimos en su operación, con la finalidad de reducir más el consumo de energía, se establecieron algoritmos, especialmente en base a operaciones matemáticas y de entrada – salida de señales especialmente análogas, que se implementaron en todas las plataformas anteriormente mencionadas (Beaglebone original, black, raspberry pi B+, banana pi); esta operación se realizó con la idea fundamental de que: si una plataforma X puede hacer operaciones de control más rápidas que una plataforma Y, las tareas encomendadas al robot que ellas controlan se harán también más rápido y de esta forma se reducirá el consumo de energía en las aplicaciones de los robots en movimiento, colaboración, etc. Los resultados en realidad arrojaron datos en donde se verifica una diferencia de operación entre las placas, presumiblemente debido a la latencia y calidad de sus componentes (no se ahondó en las razones que lo pudieran causar), pero que, en el entorno global de operación, no contribuían de forma determinante.

Concomitantemente con todo esto y, no de menor importancia, estaba el desarrollo de las tareas de colaboración entre robots, y entre ellos y la plataforma de supervisión; esto prioriza directamente la utilización de una red inalámbrica, más adelante se expondrá el estudio que se realizó, pero de forma primaria, se determinó el uso de la misma red que inicialmente se probó en el robot original (Lucy01) y que al final será determinada también como la más efectiva de forma teórica y especialmente de forma práctica.

Como se mencionó en el párrafo anterior, cuando se hicieron las pruebas de carga computacional en cada una de las plataformas, a pesar de que tenían tiempos de respuesta diferentes, estos no eran demasiado determinantes, muchas veces aparentemente se hacían en el mismo tiempo, es decir las diferencias resultaban imperceptibles, excepto cuando las aplicaciones se hacían un poco más serias como por ejemplo el desarrollo de los algoritmos de multiplicación de matrices, aquí se exigieron más a las tarjetas y si se pueden evidenciar diferencias en tiempo muy marcadas, pero nuevamente, nada que afecte de forma muy determinante en el consumo total de energía de la plataforma robótica móvil completa. Esto, y a pesar de que no se ha tenido en ningún momento un medidor real de potencia consumida, hizo que se concluya directamente que la parte motriz de la plataforma y sus comunicaciones son las que más cantidad de energía consumen y es allí en donde debían atacarse en las estrategias.

Por todo lo dicho anteriormente, tanto para las baterías, cuanto al control, cuanto a las comunicaciones, todas estas placas resultarían eficientes en su utilización, se decide entonces, construir una plataforma electrónica, que permita montar sobre ella cualquiera de las placas con un objetivo común: gobernar por sí misma y de forma autónoma un robot móvil, que tiene su hardware motriz con una estructura básica: dos motores, dos o cuatro encoders por cada rueda (según configuraciones de fábrica, no determinantes) y su fuente de alimentación, con baterías tipo LIPO, que alimente, tanto para la capa motriz como para la capa de control y comunicaciones; esta particularidad exige la regulación de tensiones, tanto a la entrada del “robot” móvil como a la entrada de la placa de desarrollo. Es decir, se construye una plataforma móvil que permite ver de forma transparente a la placa de control y ejecuta sus tareas de locomoción y comunicaciones, de forma autónoma; a esta plataforma se le denomina por su concepción: “robotGym”. El diseño del pcb de esta placa se puede observar en la Figura 3.1.

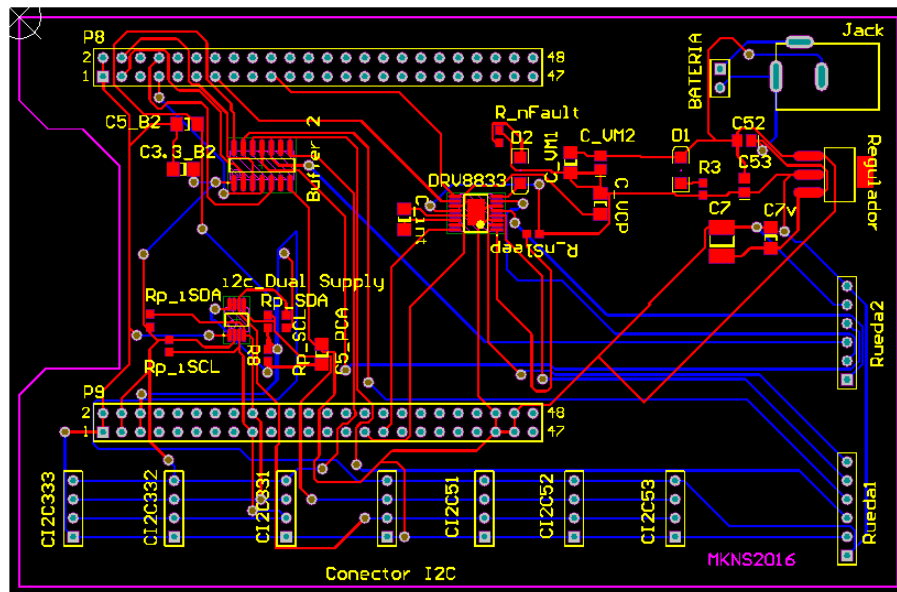


Figura 3.1. PCB de la placa de propósito de movimiento general

Como se observa en la Figura 3.1., la tarjeta está diseñada de tal forma que tenga la electrónica necesaria para: en primer lugar, proporcionar una tensión estándar regulada a valores necesarios para alimentación de los componentes electrónicos de ella misma y de la placa de control que se monte sobre ella, contiene también un CI DVR8833 [27] que es un puente H para controlar la potencia de los motores, un buffer conversor de niveles de tensión (TXB0104) [28] que permite transmutar a las tensiones de entrada a varios niveles de tensión desde las placas a niveles adecuados y viceversa; para este caso empleamos conversiones entre 5 Vdc y 3.3 Vdc; finalmente esta placa tiene también un bus bidireccional de señales I2C para las comunicaciones

[illegible]

- 34 -

La plataforma móvil robotGym además posee un zócalo común de varios pines de I/O, en donde a través de cables o jumpers se conectarán las señales provenientes de la placa de control en donde se ejecutarán, primeramente, los algoritmos primitivos de movimiento del robot; luego las lecturas de los encoders como resultado del movimiento, que permitirán detectar su posición mediante el método de odometría.

De esta forma lo que se tiene que controlar es que las placas posean los elementos básicos necesarios para conexión con éste sócalo mencionado para enviar señales analógicas en forma de PWM a los motores del robot para su movimiento, a través de un puente H anteriormente mencionado, que acople la potencia de ellos; y recibir por un GPIO las señales provenientes de los dos o cuatro encoders ópticos provistos en la plataforma, para que el programa discrimine la posición del robot, esto se lo hace únicamente con la detección de los flancos de las señales binarias de los encoders, alimentados a un contador o a un programa de conteo.

Este sistema pretende, primeramente, controlar la velocidad y maniobrabilidad del robot, para ello es necesario generar desde la placa una señal PWM (o dos dependiendo del motor utilizado) que alimente al motor de cada rueda; definiendo así una relación entre el porcentaje de variación de la señal PWM a la velocidad de cada rueda del robot (esta relación no será la misma en todos los motores, por ello se ha de utilizar métodos de corrección de la velocidad o del movimiento, pues, de no ser así, cuando se envíen señales iguales, que harían que el robot se desplace en línea recta, tal vez se tendría una ligera curvatura en cualquier dirección (derecha o izquierda).

La placa robotGym, no sirve en particular para un robot o tipo de robot específico, es para cualquier plataforma que tenga el concepto; al tomar en cuenta lo dicho en el párrafo anterior que hay errores en la odometría producidos por la diferente relación en las salidas PWM que alimentan los motores, se decide también en este “entrenador” robótico la implementación de un robot que pueda conectarse a él y que tenga la posibilidad de implementar un sistema de control de estabilidad de las velocidades en las ruedas, tal como en los robots que posteriormente en este capítulo se detallarán; y es así que se implementó una nueva plataforma que realiza este control (tipo PID) mediante la incorporación de un sistema Arduino Mega y una comunicación inalámbrica con una PC que será la que efectúe el algoritmo de control de estabilidad; la Figura 3.3 indica esta plataforma construida.

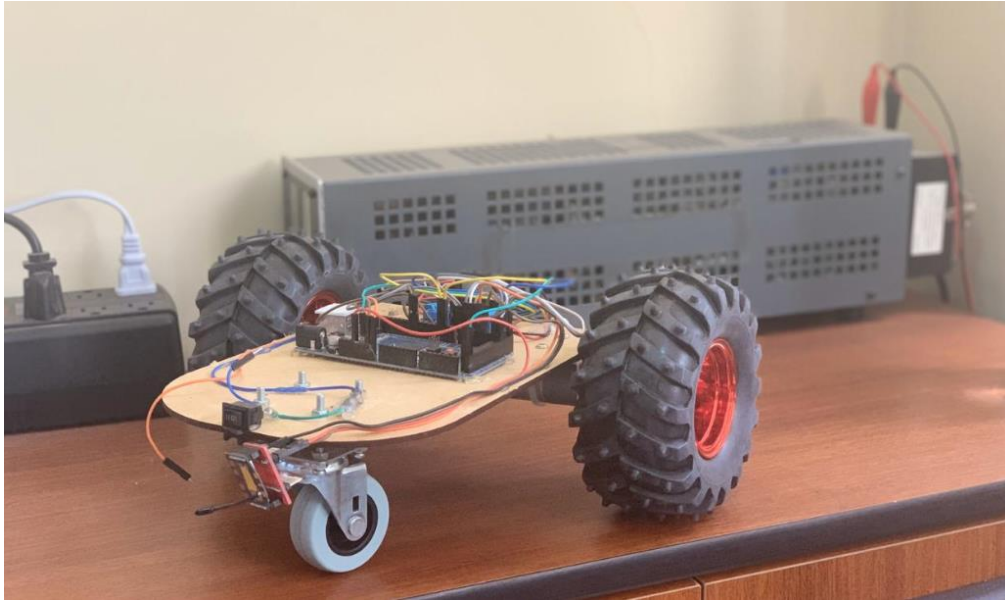


Figura 3.3. Robot entrenador con control de estabilidad.

Todo este trabajo fue evaluado en las etapas posteriores de la tesis, pues si bien es cierto las plataformas embebidas, son muy eficientes y muy versátiles, al momento de escribir los programas de aplicación en ellas, se volvieron inservibles, pues los algoritmos desarrollados, necesitaban gran capacidad de programación y usar lenguajes de alto nivel, específicamente MATLAB para aprovechar el uso de funciones matemáticas complejas, por lo que, todo este trabajo se declinó y empezó a utilizar computadores que desarrollen uno o varios (según la aplicación) programas en MATLAB, que se ejecuten en los robots de forma local, montando la computadora en ellos, o de forma remota, introduciendo dentro de los algoritmos, tramas de comunicaciones que permitan la operación eficiente, esto obviamente involucra un nuevo análisis en el caso de existir varios robots en aplicaciones de colaboración, como por ejemplo los retardos máximos de comunicaciones entre robots, la plataforma central y demás, eso se explica en el texto a continuación.

3.1 Construcción de las plataformas robóticas para aplicación.

3.1.1 Robots de tipo Uniciclo.

Para la construcción de este tipo de robots, que como se había mencionado en los párrafos anteriores, son uno de los dos tipos seleccionados para utilizarse en las pruebas de las aplicaciones planteadas para cumplir con los objetivos de esta tesis, conjuntamente con las plataformas móviles omnidireccionales, vista su versatilidad y capacidad de movilidad; es importante contar específicamente con un diseño tanto mecánico como electrónico que permita, en primer lugar, mantener esta versatilidad de los movimientos que se mencionado, como la

capacidad e inteligencia computacional para llevar a efecto todos los procedimientos matemáticos que se establecen en las aplicaciones.

a. Geometría de la estructura mecánica del robot.

Esta geometría será utilizada en algunos robots de tipo Uniciclo y Omnidireccionales, se basa en la estabilidad mecánica que debe tener un robot móvil para su perfecto funcionamiento.

La estructura mecánica de esta plataforma se desarrolla de tal manera que en ella se puedan colocar de manera efectiva y segura los componentes mecánicos, eléctricos y electrónicos que conforman el robot, delimita el espacio interno y se encarga de distribuir las cargas mecánicas en segmentos diferentes que para ese mismo fin se han diseñado y dimensionado; aquí se colocarán los ejes, los soportes, poleas, y sistemas de transmisión; así mismo como los elementos eléctricos y electrónicos y de control, como los servos, los motores, actuadores, placas embebidas, computadores, fuentes de alimentación, etc.

Para la construcción de robots uniciclo y cuatriciclo, el diseño geométrico del chasis, considera la estabilidad y resistencia de la estructura, por lo que la plataforma debe estar apoyada sobre cuatro ruedas (independientemente si las cuatro, o solamente dos de ellas tienen tracción mecánica); es por esta razón que se ha seleccionado una forma de prisma hexagonal no uniforme, principalmente por el requerimiento de estabilidad; situación muy necesaria al momento de construir un manipulador móvil sobre él, es decir la incorporación de un brazo robótico a la estructura; además se han incorporado unos segmentos inclinados en las esquinas para optimizar el peso y evitar de mejor forma los obstáculos con que se pueda encontrar en el ambiente de operación. La Figura 3.4 muestra la geometría de los robots a implementarse.

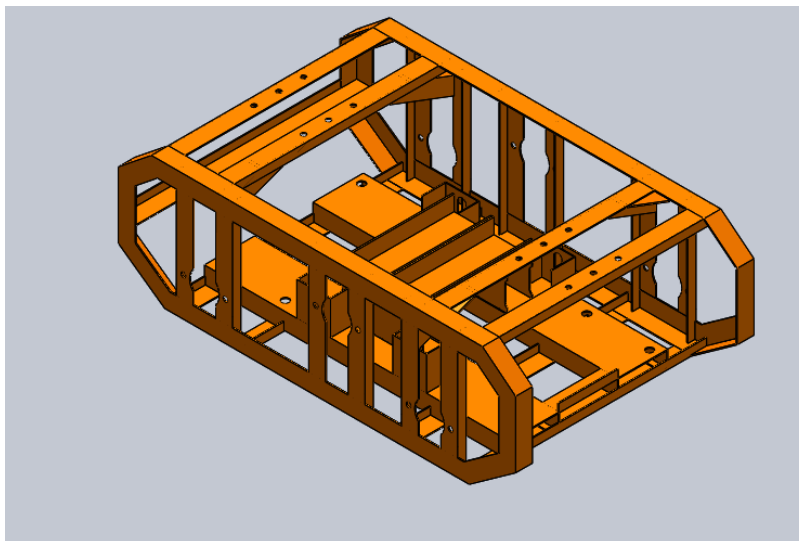


Figura 3.4 Diseño geométrico hexagonal para los robots móviles de aplicación.

Las dimensiones aproximadas de esta estructura son 60 cm de largo, 36 cm de ancho y 20 cm de altura. Para el sistema de transmisión, originalmente se diseñó y desarrolló un sistema de bandas para la conexión de los ejes de los motores y las ruedas de tracción, luego con la experiencia, se montó un sistema de cadenas y piñones con el fin de evitar los deslizamientos y la holgura en las ruedas y el eje del motor. Por esta razón es importante también considerar los espacios existentes entre las partes estáticas y en movimiento para que los actuadores y el sistema de transmisión tengan libertad para girar, es por eso que se estableció una distancia de 3 cm entre la banda y el extremo lateral de la carrocería; así mismo los espacios y soportes para la colocación de motores en el centro de la plataforma, destinando los extremos frontal y terminal para las colocación de las fuentes de alimentación y la computadora específicamente.

Esta estructura, permite además fijar en ella un brazo robótico para la construcción y aplicación en ella de un robot manipulador móvil.

En el diseño mecánico de la plataforma se han tomado como referentes fundamentales: que esté compuesta por una estructura compacta y resistente que permita soportar y transportar todos sus componentes internos ya descritos de forma general, al mismo tiempo que brinde estabilidad y confiabilidad en los movimientos; los siguientes aspectos son las consideraciones generales de necesidades fundamentales: La **capacidad de carga** que está definida inicialmente por los componentes internos, es decir la computadora de a bordo, los motores, los actuadores sobre ellos, las fuentes de alimentación y las placas de acoplamiento; y después la incorporación de un robot manipulador para aplicaciones específicas, allí se toma en cuenta las características particulares y el diseño de este tipo de robots; la **velocidad** para transportar objetos en aplicaciones preestablecidas, con una velocidad media aproximada de 0.88m/s; la **configuración mecánica** de la plataforma determinada por el volumen interior, que debe incluir los elementos internos. Adicionalmente, la construcción y aplicación de los robots pueden incluir el uso de orugas, piernas mecánicas o ruedas; todas las aplicaciones hechas necesitan de ruedas para la locomoción de los robots, esto porque son mucho más fáciles de controlar y la forma de medición de los desplazamientos de ellas con respecto a los de la plataforma completa es mucho más intuitiva; son más rápidos, más eficientes en el uso de la energía y admiten mayores cargas por el soporte adicional de goma solamente o goma y aire.

La necesidad fundamental es de una estructura fija, liviana y estable, es por esta razón que se selecciona ángulos de aluminio de aletas iguales de 25.4mm y 0.213Kg/m de forma de obtener mayor firmeza en la estructura y mejor soldadura entre partes; se usa esta característica en todo el perfil externo de la estructura y para reforzamiento de las paredes verticales se usa un perfil de peso inferior (0.139 Kg/m) y para la base de la estructura un perfil de 0.56 Kg/m pues debe soportar el peso interno de los elementos internos (los de mayor peso son las baterías).

b. Cálculo de cargas en la plataforma robótica móvil.

Los objetivos de la presente investigación se centran en la construcción de una plataforma móvil que resuelva los problemas de localización y búsqueda específicamente, y dentro de ellos también, la reducción del consumo de energía de los robots móviles en tareas particulares o colaborativas y la implementación de una red de comunicaciones eficiente y versátil.

Inicialmente en la plataforma robótica móvil necesita considerarse únicamente a sí misma y posteriormente a la incorporación de objetos o elementos adicionales, pues al ser un entorno colaborativo, las tareas de colaboración pueden ser rescate, transporte, búsqueda, etc., para lo cual será susceptible la incorporación de brazos robóticos, hojas o cuchillas de limpieza y arrastre, cilindros de soporte de transporte, antenas, etc.

Por lo dicho el peso total a considerarse en el robot será la suma de sus pesos individuales: W_p peso de la plataforma; W_b peso del brazo; W_g peso de la pinza o mordaza; W_l peso de la carga; W_s peso del sensor y W_T peso total.

$$W_T = W_p + W_b + W_g + W_l + W_s + 10\%W_T \quad (3.1)$$

c. Dimensionamiento de los motores.

Por la naturaleza de la plataforma, tipo de movimiento, control y velocidad, se debe considerar previamente la potencia y la relación de transmisión:

$$\begin{aligned} P_{req} &= F \cdot \vec{V} \\ P_{req} &= 320N(0,8 \frac{m}{s}) \\ P_{req} &= 255w \\ P_m &= \frac{P_{req}}{i_r} \\ P_m &= 515w \end{aligned}$$

Con la referencia de la potencia calculada en el párrafo anterior se deberán escoger entre una amplia gama de motores que permitan esta capacidad, sean versátiles, eficientes y económicos; se selecciona los modelos Dynamixel PRO H54-200-S500R, pues estos motores además de sus características espectaculares, que se indicarán posteriormente en el diseño electrónico, posee una entrada directa a un controlador PID al actuador, lo que hace que su eficiencia y la de los controles de posición del robot sean mejoradas en un porcentaje sumamente alto con relación a motores convencionales de cc o ac. Se comprueba la operación del robot tomando en cuenta las constantes y características de sus motores y el diseño:

$$\begin{aligned}
P_m &= 2 * 200w = 400w \\
\omega_m &= 27.9 \text{ rev} / \text{min} = 2.92 \text{ rad} / \text{s} \\
T_m &= 44.2 N * m \\
P_{M1} &= \omega * T_m = 129,5w \\
P_T &= P_{M1} * 2 = 258.3w
\end{aligned}$$

Que resulta la potencia máxima que los motores pueden generar.

d. Velocidad Requerida.

La velocidad en una plataforma robótica móvil de este tipo, que se usa para aplicaciones exigentes, no debe ser muy rápida porque la relación entre la rapidez y la precisión debe ser muy baja, además de la seguridad que necesita el robot en sus movimientos, por lo que la velocidad necesaria sería:

$$\begin{aligned}
\overrightarrow{V_{req}} &= 0.8 \text{ m} / \text{s} \\
\overrightarrow{V_i} &= \frac{P_T}{F_{req}} = 0.807 \text{ m} / \text{s} \\
\overrightarrow{V_{req}} &= \frac{\omega_{req} * \phi_{rueda}}{2} \\
\omega_{req} &= \frac{2\overrightarrow{V_{req}}}{\phi_{rueda}}
\end{aligned}$$

e. Diámetro de las ruedas.

El diámetro requerido de las ruedas para mejor acoplamiento de los motores Dynamixel es de 26cm (tamaño estándar sugerido por el fabricante). Por su contexto de soporte de mayores pesos en la plataforma son ruedas inflables de 26cm; por lo tanto:

$$\begin{aligned}
\omega_{req} &= \frac{V_{req}}{R_{rueda}} \\
\omega_{req} &= 6.15 \text{ rad} / \text{seg}
\end{aligned}$$

f. Relación de transmisión.

Para tener una referencia que especifique la diferencia entre la velocidad angular requerida y la generada, se usa este índice de relación:

$$i = \frac{\omega_{req}}{\omega_{mot}}$$

$$i = 2.08$$

Esto nos indica que es necesario tener un sistema de transmisión que entregue la velocidad requerida para el tiempo y diámetro de la rueda seleccionada a plena capacidad de inflado.

g. Poleas.

El suministro de velocidad al robot y sus características motrices, no dependen únicamente de los motores, sino también de las poleas que arrastren a las ruedas delanteras, es por eso que es necesario encontrar los diámetros exactos de las poleas de transmisión entre el eje de las ruedas y el eje del motor, para obtener la velocidad angular de la plataforma se multiplica la relación de los ejes por la velocidad angular del motor dando como resultado:

$$i_{real} = \frac{N_1}{N_2} = 2.14$$

$$\omega_{real} = i_{real} * \omega_{motor} = 6.25 rad / s$$

Con este valor y el diámetro seleccionado de las ruedas, se obtiene la velocidad lineal de robot, que es de:

$$\vec{V} = 0.812 \frac{m}{s}$$

Como se observa la velocidad calculada devenida del sistema de transmisión coincide con la requerida para el robot móvil, consumiendo una potencia de 259.84 w que es menos a los 400w del motor.

h. Cálculo de los ejes.

Los ejes del robot estarán expuestos a fuerzas tangenciales, cortantes y de rotación, es necesario tomarlas en cuenta para el diseño adecuado del robot; además de que se debe considerar que una banda conectará el eje y que está sometida a tensión, así como también se debe considerar el peso total de la plataforma. Es necesario entonces un análisis de esfuerzos a los que está sometido el sistema; dicho análisis se adjunta en los anexos respectivos, dando como resultado que el diámetro necesario del eje es de 153mm, tal como se aprecia en la Figura 3.5.

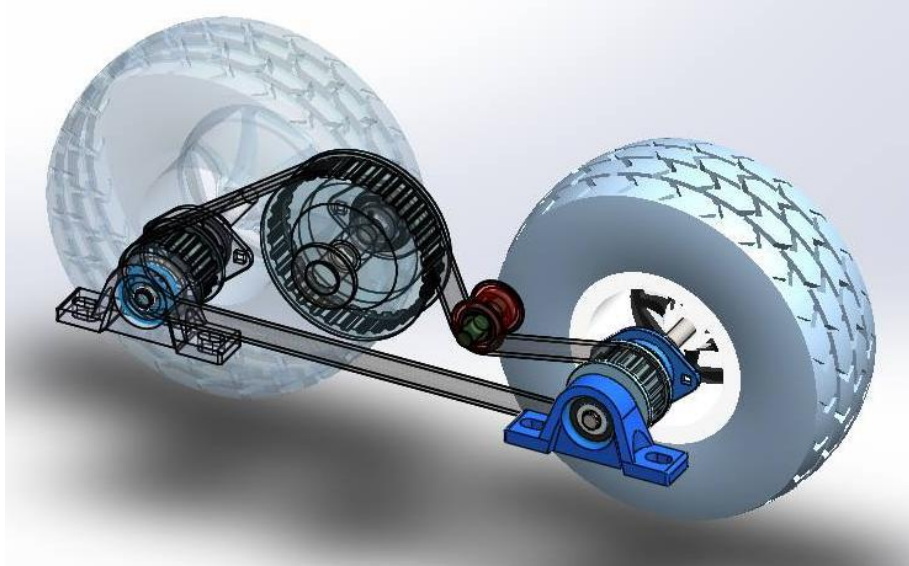


Figura 3.5 Sistema de transmisión de giro del motor a las dos ruedas motrices.

3.1.2 *Análisis estructural.*

Utilizando software de simulación se realiza la prueba de comportamiento de la estructura del robot, la cual es sometida a un tipo de esfuerzo específico en donde se cree las partes más vulnerables del robot, por ello primeramente se malla el diseño y define dichos puntos críticos; El análisis de tensión es realizado por partes de forma de que se individualicen los componentes de la estructura como elementos sólidos vacíos de tipo viga para proceder a calcular las tensiones o las deformaciones que como respuesta a las fuerzas y presiones, presenta la estructura, este resultado se verifica en la Figura 3.6.

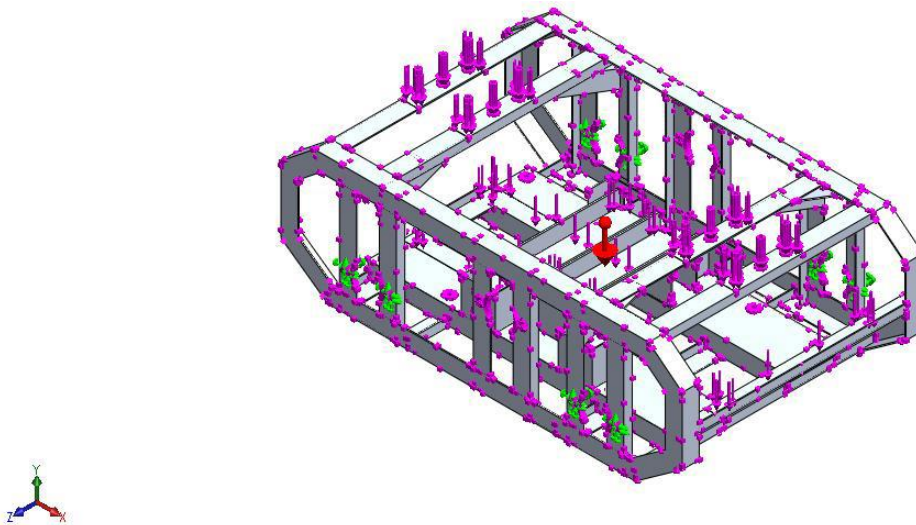


Figura 3.6. Simulación de fuerzas aplicadas a la estructura de soporte.

Se utiliza para análisis de resultados, criterios de falla elástico, los cuales no deberían sobrepasar el valor máximo de 59.23 MPa; pues se va a comparar el valor máximo de resistencia del elemento de aluminio utilizado correspondiente. Los resultados se observan en la Figura 3.7.

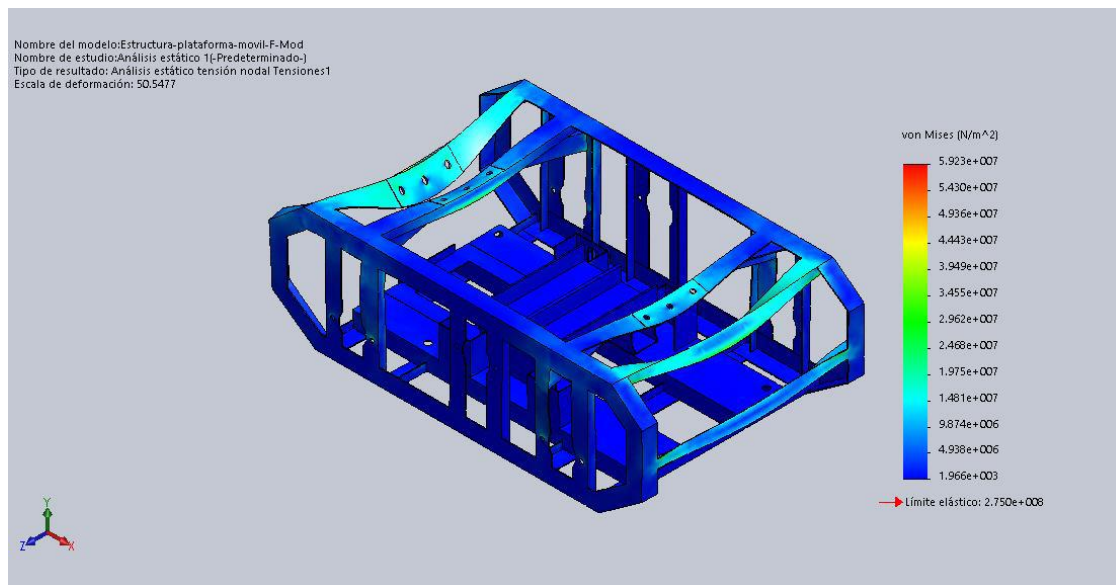


Figura 3.7 Resultados de simulación para el análisis de la tensión de la estructura

Para finalizar el análisis se puede observar que en la parte considerada más débil de la plataforma se han aplicado esfuerzos característicos de tal forma que se pueda verificar su factor de seguridad que debe ser mayor que 1; La simulación demuestra que la estructura superará esas cargas aplicadas; esto se observa en la Figura 3.8.

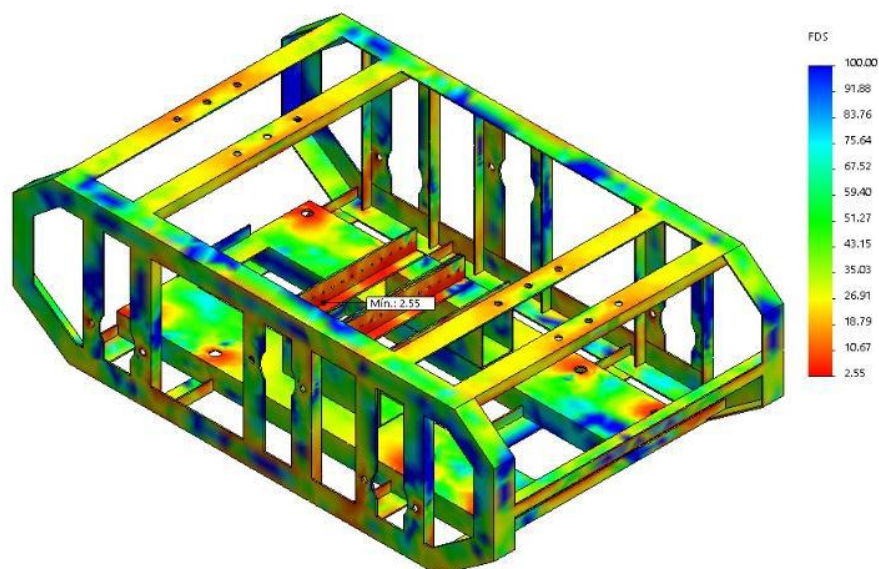


Figura 3.8. Resultados de la simulación para verificar el factor de seguridad.

Como se observa el diseño de la geometría de la plataforma es el adecuado considerando parámetros involucrados como: peso, colocación y organización de los elementos eléctricos, electrónicos y mecánicos, además que proporciona amplia capacidad de maniobrabilidad y movimiento; así como también resistencia y estabilidad. Los elementos inclinados de ella se usan para optimizar el peso y evitar obstáculos de forma eficiente.

a. Diseño de los ejes.

Se consideran las fuerzas aplicadas a ellos, de naturaleza tangencial, cortante y rotacional, en este tipo de robot existen dos ejes, los del motor y los de las ruedas.

1. Eje del Motor.

Es el eje principal que transmite el movimiento desde el actuador hacia la polea que se conecta de forma opuesta al otro extremo de una polea que está engranando la banda de transmisión.

2. Eje de la rueda.

Para este caso se han diseñado con distintos diámetros; pues es afectado por varios elementos mecánicos, como las chumaceras (primer nivel), la polea de transmisión (segundo nivel) y la chumacera de la pared del robot, que conecta a un bocín que al final es quien da movimiento a la rueda. (el diseño mejorado, previene el desplazamiento de la banda, colocando un sistema de cadenas y engranajes, esto previene este deslizamiento especialmente por efecto del shaft del motor).

3. Tensores.

Son los elementos mecánicos que permiten regular la tensión que presenta la banda; son diseñados de forma cilíndrica y de acuerdo a las dimensiones de la banda y los rodamientos internos y externos.

3.1.3 Diseño del sistema eléctrico y electrónico.

Dentro del sistema robótico, la otra etapa fundamental es su operación electro mecánica; bien es sabido que este tipo de robots funcionan girando dos o más ruedas motrices en ellos, alimentando motores de corriente continua desde baterías de larga duración; los suministros de parámetros eléctricos a estos motores dependen de los movimientos que el robot vaya a realizar, es decir si va en línea recta, describe curvas, etc. Es necesario entonces un sistema eléctrico que manipule la electricidad que va a ser entregada por las baterías, que son un elemento fundamental y hasta neurálgico en el sistema, pues de dejar de operar por cualquier razón, toda la plataforma queda inútil; de ahí que las aplicaciones se centran en esta relativa debilidad que presentan dichos robots.

Se debe entonces considerar su capa de alimentación (y control de ella) y su capa inteligente, es decir el hardware electrónico de control y medición de características inherentes a la operación fundamental del robot en primer término y luego en uso de sensores y aplicaciones específicas; de lo dicho hay entonces tres aspectos en el campo eléctrico a considerar: la alimentación, el control y las aplicaciones.

a. Sistema de alimentación.

Se requiere una alimentación de 12 Vdc para la computadora a bordo y los sensores a ser utilizados en las aplicaciones principales y específicas. Es necesario, además, un circuito de control que permita controlar la tensión en los actuadores (que generaran el movimiento deseado del robot móvil) y un sistema que mida los valores de las variables de importancia en el sistema (como los encoders y demás).

El primer sistema requerido es el sistema de control de la alimentación de los elementos mencionados; consta de varios elementos como protecciones eléctricas, alimentación de motores, medición de corriente y presencia de las baterías, un reductor de voltaje para alimentación de dispositivos de baja potencia, visualización de la información y el controlador electrónico del sistema. En la Figura 3.9 se puede observar el diagrama del circuito usado para lo mencionado.

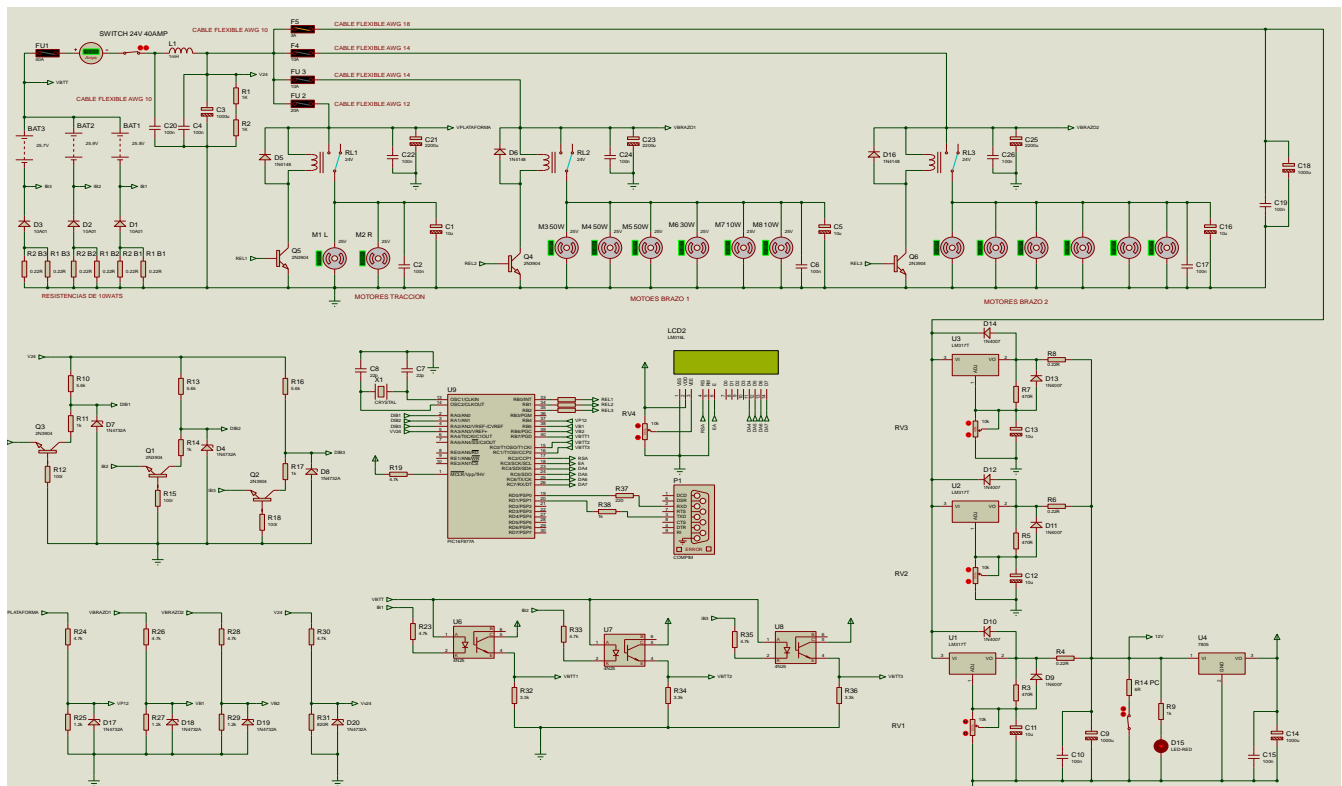


Figura 3.9. Circuito de control de alimentación y etapas.

b. Etapa de potencia.

Consta de 5 fusibles: uno principal de 40 A, que se encarga de la protección del circuito completo, otro fusible de 20 A para proteger los motores del robot móvil, dos fusibles de 10 A para protección extra de motores y acoplamiento del brazo para aplicaciones especiales; este está diseñado por si se emplean o acoplan dos brazos robóticos en la plataforma. Y un fusible de 3 A para protección del ordenador a bordo y los demás elementos electrónicos que se alimentan desde estas baterías centrales. Se observa en la Figura 3.10.

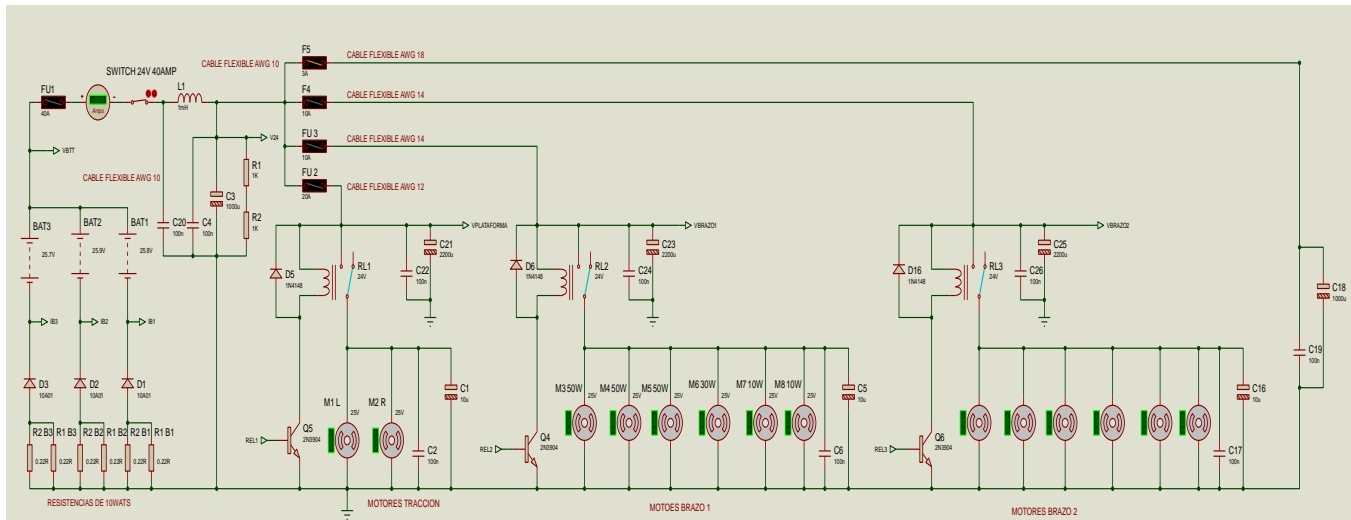


Figura 3.10. Circuito de protección y potencia.

Este circuito arriba visualizado, contienen además 2 diodos para impedir que las baterías se descarguen entre sí cuando haya pequeñas diferencias de tensión, así como también, relés que desconectan la alimentación entre la plataforma móvil y el (los) brazos robóticos que se utilizan en las aplicaciones; estos relés se accionan desde un microcontrolador particular. Se usa igualmente varios capacitores que amortiguan las sobrecorrientes y protegen a los motores.

c. El reductor de voltaje.

Este circuito particular, alimenta al computador y a otros dispositivos como sensores y cámaras: La disminución del voltaje se lo realiza a través del uso de tres CI reguladores de tensión LM317, que se configuran para regular a 12 V y conectan en paralelo para permitirles tolerar mayores valores de corriente, luego se reduce el voltaje a 5 Vdc que es usado para la alimentación del microcontrolador y otros elementos auxiliares. Su diagrama se muestra en la Figura 3.11.

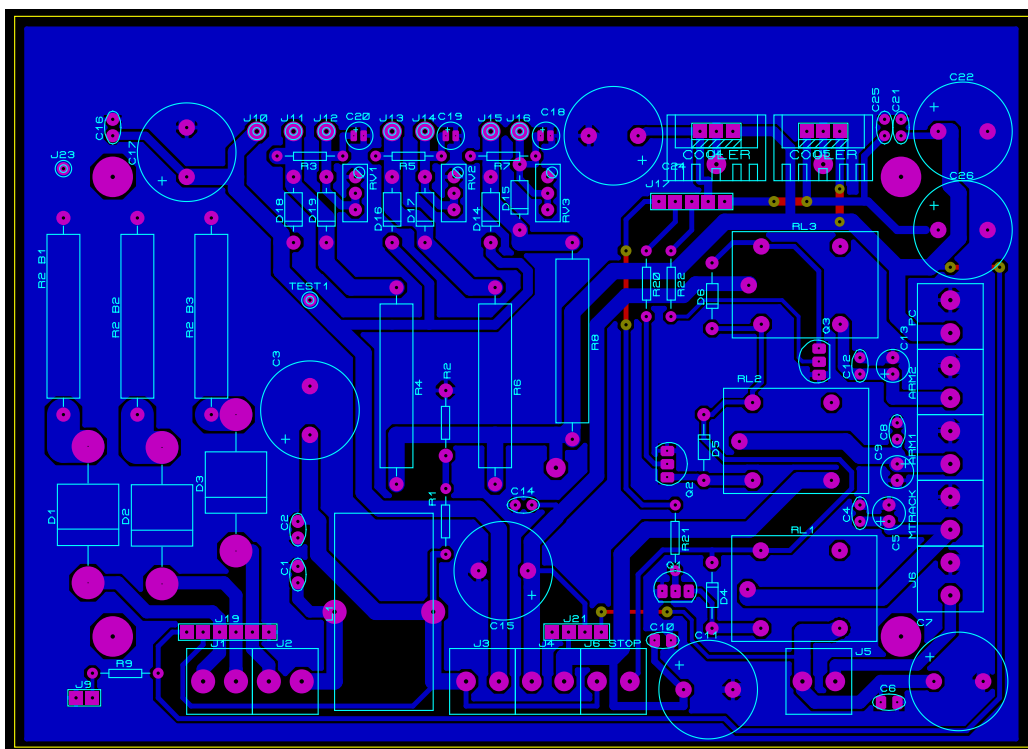


Figura 3.13. (a) PCB de la Placa de Convertidor DC/DC

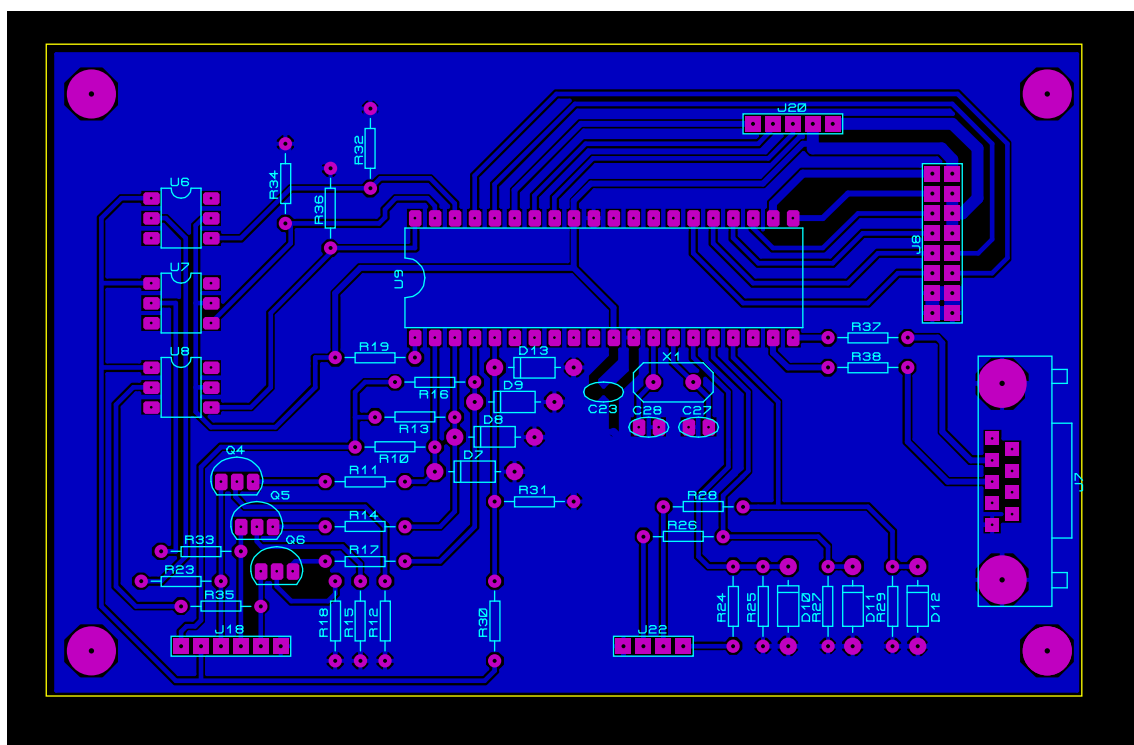


Figura 3.13 (b) PCB de la placa de protección de baterías LIPO

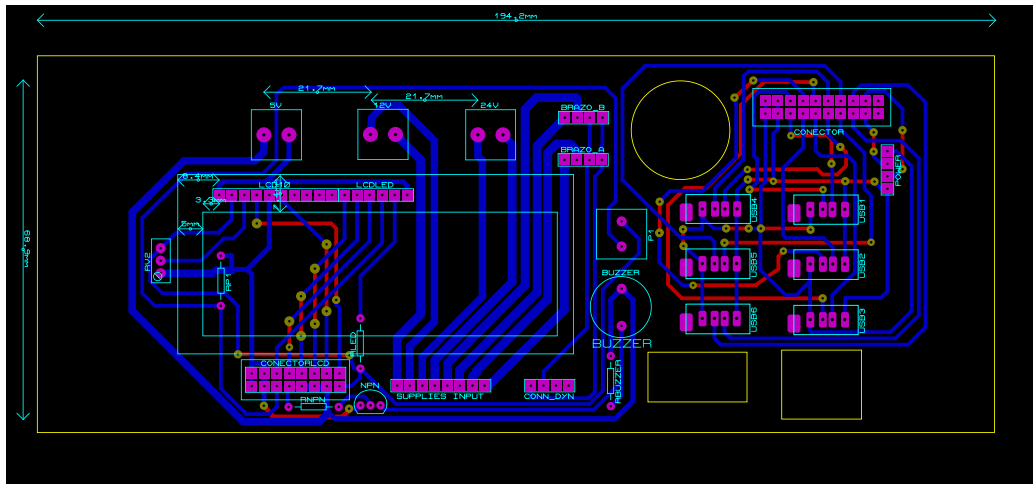


Figura 3.13 (c) PCB de las placas de conexión de periferia.

f. Implementación de los circuitos electrónicos.

Luego de rutear las placas y ensamblar los elementos se construyen las placas electrónicas detalladas, se procede a realizar las pruebas de continuidad, el soldado y colocación y pruebas de elementos electrónicos y la implementación y acoplamiento en la plataforma.

Los resultados se muestran en las Figuras 3.14 (a), (b) y (c).

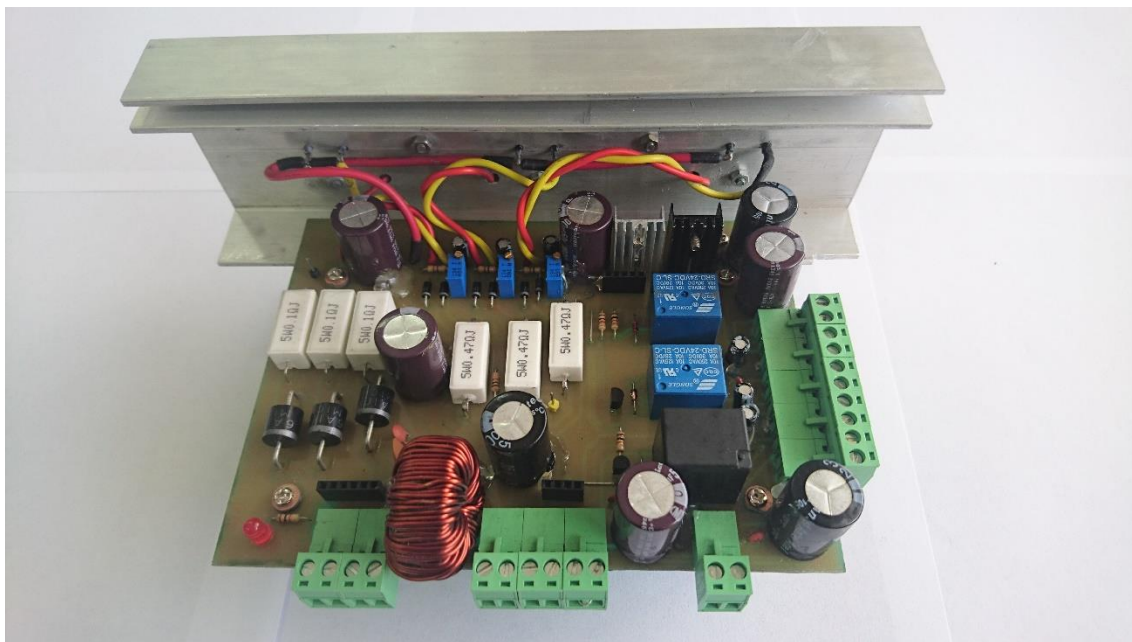


Figura 3.14 (a) Convertidor DC/DC

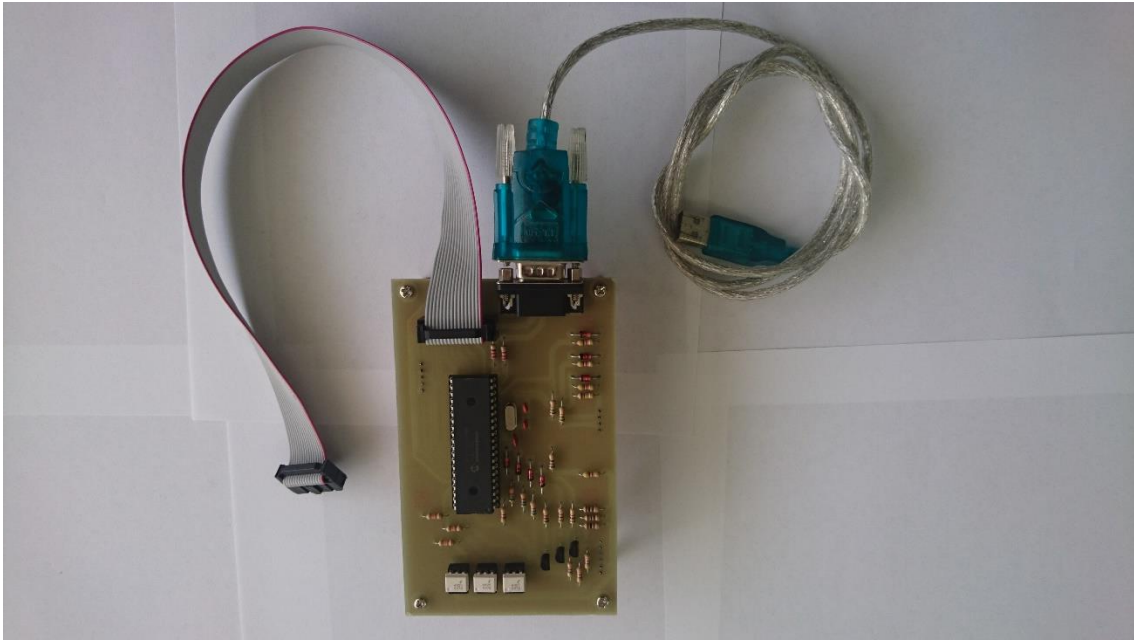


Figura 3.14 (b) Protección de baterías LIPO

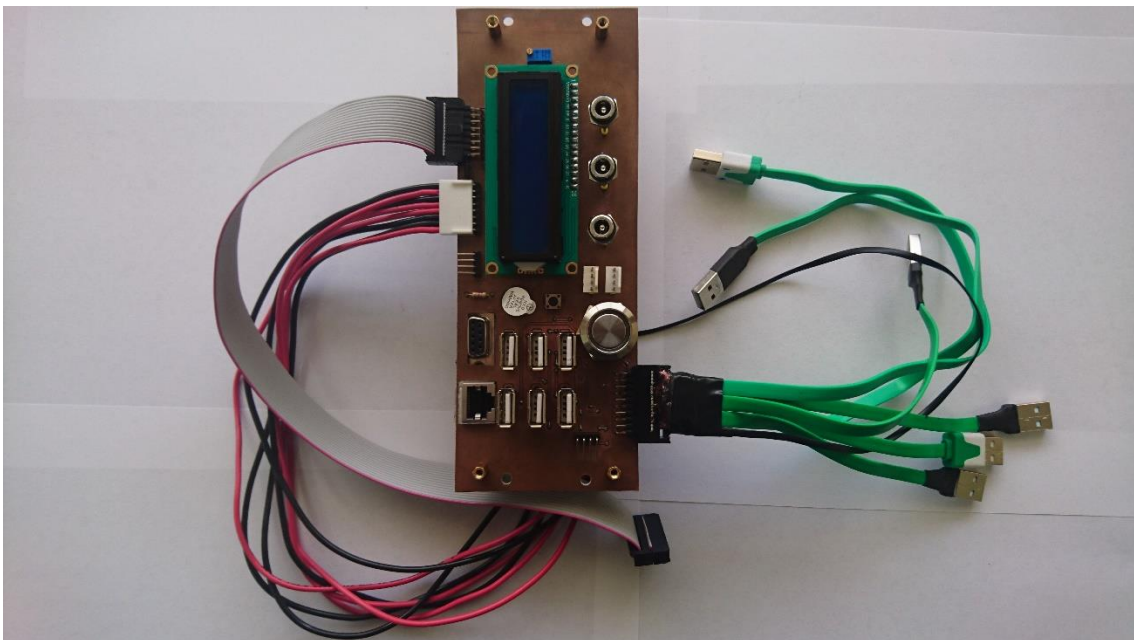


Figura 3.14 (c) Extensor de periferia central

Para estar seguro de las operaciones efectivas dentro del robot, se prueban las placas para las peores condiciones de operación respondiendo estos de forma segura y completamente confiable además de presentar una respuesta a sobrecargas más rápida de la esperada.

3.1.4 Robots de Tipo Omnidireccional.

Cuando se trata de robots omnidireccionales, una de las cuestiones importantes es: ¿de cuántas ruedas se compone? Ya que podría tratarse de robots de tres ruedas, cuatro ruedas o más. Cada tipo tiene ventajas y desventajas asociadas como la estabilidad, tracción, control y maniobrabilidad. En este caso, vista la estructura similar y las aplicaciones que se desarrollaron, se utilizan robots de cuatro ruedas.

Los robots omnidireccionales permiten a su vez el uso de más tipos de ruedas y su disposición, es decir su estructura pueda ya no ser solamente un rectángulo o un cuadrado, sino también circular u ovalada, esto también dependerá muchísimo de las aplicaciones orientadas (se va tomando conciencia oportuna de que no existe un solo tipo de robot ideal, sino más bien que su uso será el que determine su característica específica). También existen alternativas para el uso de este tipo de ruedas omnidireccionales, repito en una plataforma circular usando ruedas de tipo universal, de ruedas orientables (algo así como las ruedas locas, pero con tracción) y de tipo mecanum.

En los robots omnidireccionales de tipo mecanum las ruedas se colocan de la misma forma en las que se ubicaran en un automóvil, con un cierto ángulo de rotación en sus rodillos que generalmente es 45° con referencia a la circunferencia exterior de la rueda; la precisión y maniobrabilidad de la plataforma dependerá en mucho grado del material del que estén hechos los rodillos de las ruedas, pues de este depende la adherencia que la rueda tenga con el piso con el que tenga contacto, además de que si la superficie de trabajo del robot no es plana por completo, puede presentar muchos problemas en su operación, esto se puede corregir, incorporando un tipo de sistema de amortiguación, que además posee un conjunto de barras de suspensión.

La ventaja fundamental de los robots omnidireccionales, es su máxima maniobrabilidad, pues se puede dirigir en cualquier dirección, sin realizar rotaciones previas o demasiado grandes, esto le permite acceder a lugares que otro tipo de plataformas no pueden llegar; además de que muchas aplicaciones exigen transitar en caminos angostos y muy delimitados, especialmente en el entorno industrial.

Tal como en el caso anterior, la estructura del robot, juega un papel determinante, pues de ella dependerá las ruedas en el chasis, pero así mismo se debe considerar varios parámetros indispensables del entorno y la operación en sí mismo del robot para su elección adecuada, parámetros tales como: la aplicación, es decir el tipo de trabajo que va a realizar, el terreno sobre el que se va a desplazar, su capacidad de carga, el consumo de energía, la complejidad de la implementación, el costo e inversión, los mecanismos de control, etc.

Realizando un análisis exhaustivo de las ventajas y desventajas que presentan las alternativas de construcción de robots móviles, la maniobrabilidad es la característica más deseada para las aplicaciones en el entorno general, y es precisamente la característica más relevante del robot omnidireccional, dándole la capacidad de realizar cualquier movimiento en cualquier dirección, la capacidad de rotar en el mismo lugar sin a ver girado previamente supone un ahorro importante de tiempo y de la energía consumida y también el desgaste menos significativo de los componentes móviles, estas consideraciones son muy importantes a la hora de realizar una evaluación de las plataformas, porque en realidad el trabajo final u objetivo lo cumplen con o sin limitaciones todas las plataformas, pero el robot omnidireccional lo hace ahorrando los recursos mencionados.

a. Diseño mecánico de la plataforma omnidireccional.

Por las consideraciones de carga, resistencia, maniobrabilidad, capacidad de evasión de obstáculos, espacio interior para elementos mecánicos y electrónicos, se utiliza el modelo base de la plataforma robótica explicada en el numeral anterior, usando los nuevos parámetros de diseño:

La carga que la plataforma se establece en relación a la carga que puede soportar cada rueda, por especificaciones del fabricante, este valor se establece en 37,5 kg, de tal forma que se consideró un valor de carga total de 150 Kg, es la razón por la que se asume como consideración de diseño la masa de 100 Kg, es decir 981N.

Todos los demás parámetros mencionados en la sección 3.3.1. son considerados, pero al ser un robot más efectivo la velocidad de operación se considera de 0.6m/s. A diferencia del robot anterior ese contará con un motor conectado a cada una de las cuatro ruedas.

Considerando la Ec. (3.1) el peso total de la plataforma (lo que puede cargar más el peso de los elementos internos) es de 44Kg.

$$W_d = W_T + W_c * f_{carga}$$

$$W_d = 44 \text{ kg} + 12 \text{ kg} * 0.1 = 45.2 \text{ kg} \approx 45 \text{ kg}$$

$$W_d = 45 \text{ kg} * 9.81 \frac{m}{s^2} = 441.45 \text{ N} \approx 442 \text{ N}$$

Al ser el peso de diseño propuesto de 981 N se comprueba que el peso del diseño calculado es menor, entonces estamos trabajando dentro de los parámetros normales, en la Figura 3.15 de muestra las constantes consideradas para estos cálculos.

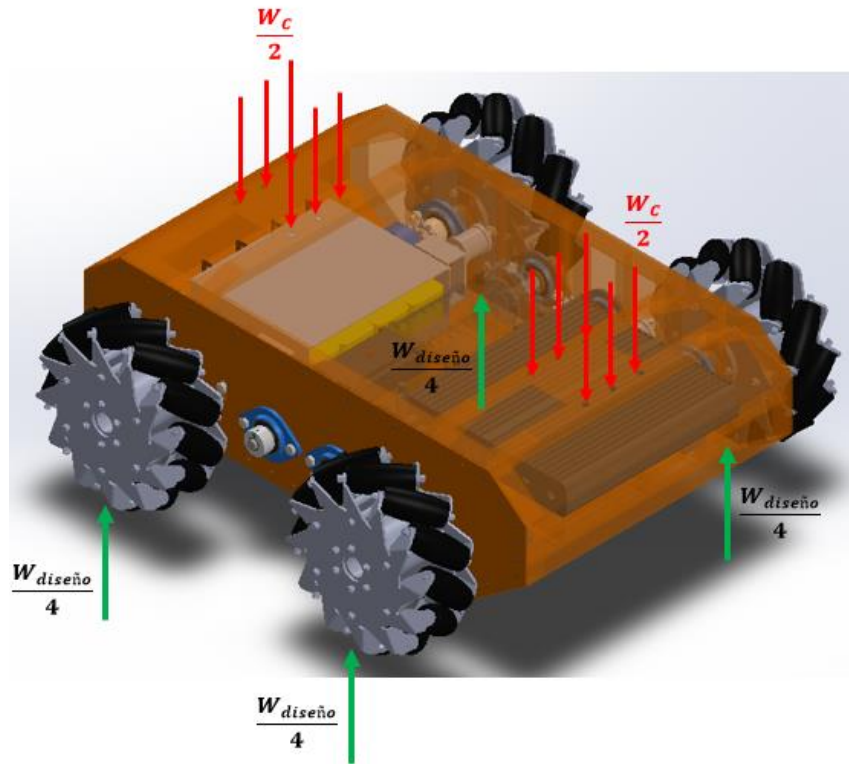


Figura 3.15 Cargas consideradas a la plataforma robótica.

b. Cálculo del Torque del sistema.

El sistema emplea un método mejorado de arrastre de la rueda, con respecto al sistema de bandas que se había propuesto en el tema anterior, es así que este robot usa un sistema de cadenas para la transmisión del movimiento del actuador a la rueda. En la Figura 3.16 se observa el sistema empleado.

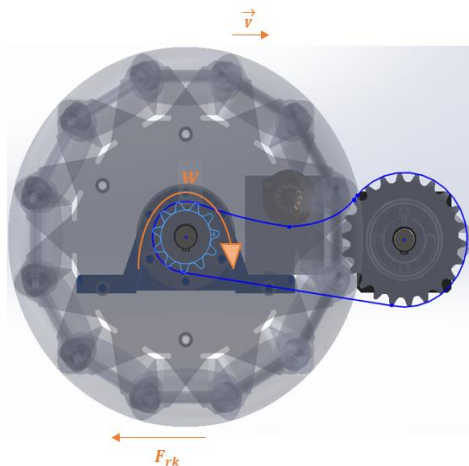


Figura 3.16 Sistema de transmisión utilizando cadenas en los ejes.

$$T = F_{rk} * \frac{\phi_{llanta}}{2}$$

$$T = 784.80 \text{ N} * \frac{0.2032 \text{ m}}{2} = 79.75 \text{ Nm}$$

$$T_{rueda} = \frac{T}{4} = \frac{79.75 \text{ Nm}}{4} = 19.95 \text{ Nm}$$

Usando estos resultados se puede calcular la potencia requerida para los motores de cada una de las ruedas motrices de la plataforma:

$$P = T_{rueda} * \omega_{requerida}$$

$$\omega_{requerida} = \frac{\vec{V}_{requerida}}{R_{llanta}}$$

$$\omega_{requerida} = \frac{0.60 \frac{m}{s}}{0.1016 \text{ m}} = 5.91 \frac{rad}{s}$$

$$P = 19.95 \text{ Nm} * 5.91 \frac{rad}{s} = 117.81 \text{ W} = 0.16 \text{ HP}$$

El motor que se requiere según estos resultados es uno que entregue una potencia nominal de 118W, de acuerdo a esto se puede utilizar los mismos motores Dynamixel PRP H54-200-S500 que satisfacen los requisitos encontrados (recordar que su potencia nominal es de 200W), además que este robot posee actuadores accionados por red, eso quiere decir que cada uno tiene una dirección única válida, que se puede configurar dinámicamente, facilitando así el trabajo de la plataforma omnidireccional, en la cual los movimientos individuales de los motores son determinantes para el movimiento conjunto de toda la plataforma.

Este trabajo se lo hace pues el motor Dynamixel posee un microcontrolador incorporado, que maneja de forma eficiente el servo, de tal manera que se supervisa su operación y en casos determinados puede sacarle de operación, los parámetros vigilados son configurables y monitorean: temperatura, tensión, par motor, etc. La potencia de cada motor determina la potencia total de la plataforma:

$$P_{M1} = \omega_{motor} * T_{motor}$$

$$P_{M1} = 2.92 \frac{rad}{s} * 44.2 \text{ N} * m = 129.15 \text{ W}$$

$$P_{total} = 4 * P_{M1} = 4 * 129.2 \text{ W} = 516.80 \text{ W}$$

c. Sistema de transmisión.

Como se mencionó antes, este robot mejora las características del robot unicycle que se había construido, en el sentido fundamental del sistema de transmisión de la tracción hacia las ruedas desde los actuadores del motor Dynamixel. Se usa para la plataforma omnidireccional un sistema de cadena, piñón y catalina; considerando la potencia transmitida de 0.174hp y una velocidad del piñón de 30rpm, se tiene opción de elegir una cadena tipo RS35 de una o dos hileras (se elige la de dos hileras por mayor seguridad y tracción, además que permite un piñón de menor diámetro y menor número de dientes).

Con la consideración de que la cadena es tipo RS35-2 a 50rpm con un paso de 3 octavos de pulgada tiene dos hileras, su capacidad de potencia aumenta y se debe considerar el factor de múltiples hileras, por lo tanto: la potencia de diseño es: $P_d = 0.16 \text{ hp} * 1.7 = 0.272 \text{ hp}$.

La relación de transmisión es:

$$i = \frac{\phi_{req}}{\phi_{llanta}}$$
$$i = \frac{0.41 \text{ m}}{0.2032 \text{ m}} = 2.02 \approx 2$$

El número de dientes de la catalina:

$$i = \frac{Z_2}{Z_1}$$
$$Z_2 = i * Z_1$$
$$Z_2 = 2 * 11 \text{ dientes} = 22 \text{ dientes}$$

donde Z_1 es el número de dientes del piñón y Z_2 el número de dientes de la catalina; los diámetros interiores, exteriores y de paso se calculan en el anexo B y no se les considera importantes para este trabajo.

La velocidad de la cadena es:

$$V_{cadena} = \frac{p * Z_1 * n}{1000} \left[\frac{\text{m}}{\text{min}} \right]$$
$$V_{cadena} = \frac{9.53 \text{ mm} * 11 \text{ dientes} * 60 \text{ rpm}}{1000} = 6.28 \left[\frac{\text{m}}{\text{min}} \right] = 20.60 \text{ ft/min}$$

dónde;

p: paso de la cadena en [mm]

n: Velocidad de salida en [rpm]

que proporciona la magnitud de las velocidades reales:

$$\omega_{salida} = \omega_{motor} * R = 27.9 \text{ rpm} * 2 = 55.80 \text{ rpm} = 5.84 \text{ rad/s}$$

$$V_{real} = \frac{\omega_{salida} * \phi_{llanta}}{2} = \frac{5.84 \frac{\text{rad}}{\text{s}} * 0.2032 \text{ m}}{2} = 0.59 \text{ m/s}$$

d. Análisis estructural.

Se considera un análisis estático de los elementos críticos dentro del robot, como los ejes de las ruedas, del motor y la estructura base que soporta la colocación de todos estos elementos, se hace una simulación y se los compara con resultados calculados.

De igual manera que en caso anterior se hace una simulación de las tensiones que se pueden presentar y aplicar en la plataforma tanto en los brazos de transmisión como en la estructura en sí y se comprueba con los valores empleados que han sido muy extremos, la plataforma cumple con las necesidades básicas para el robot; la Figura 3.17 muestra los resultados de los ensayos en la simulación.

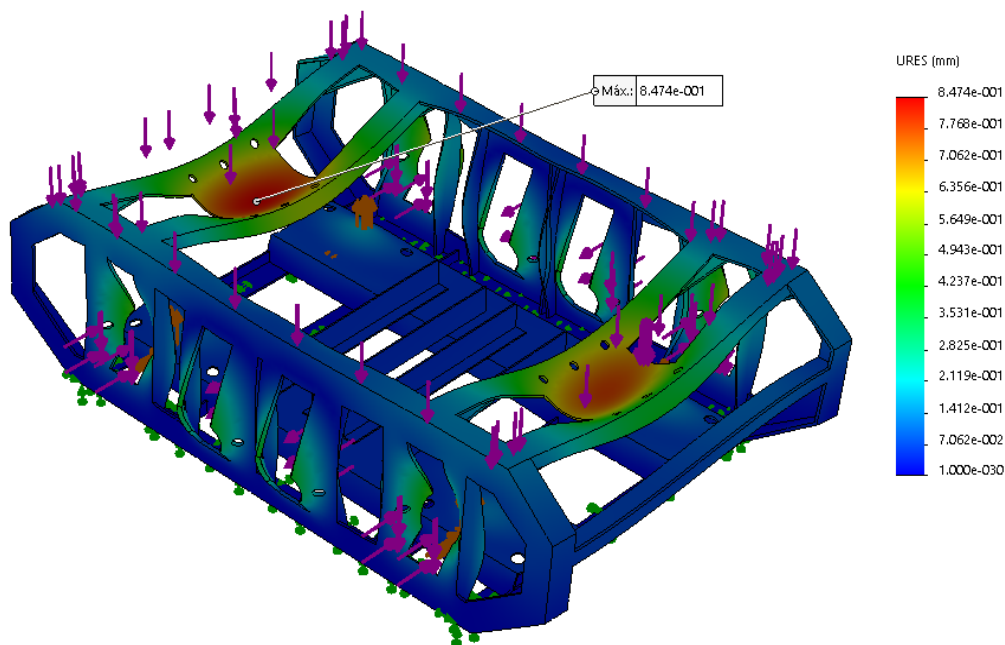


Figura 3.17 Desplazamientos resultantes a las fuerzas aplicadas a la estructura del robot.

Los mayores desplazamientos han sido considerados en la parte superior en donde se supone se colocarán el o los brazos robóticos para la construcción del manipulador móvil, que servirá para comprobar su aplicación particular y el uso de los algoritmos para ella desarrollados.

De igual forma se aplican esfuerzos a la plataforma para comprobar el factor de seguridad de respuesta que ella presenta: la Figura 3.18 muestra los resultados.

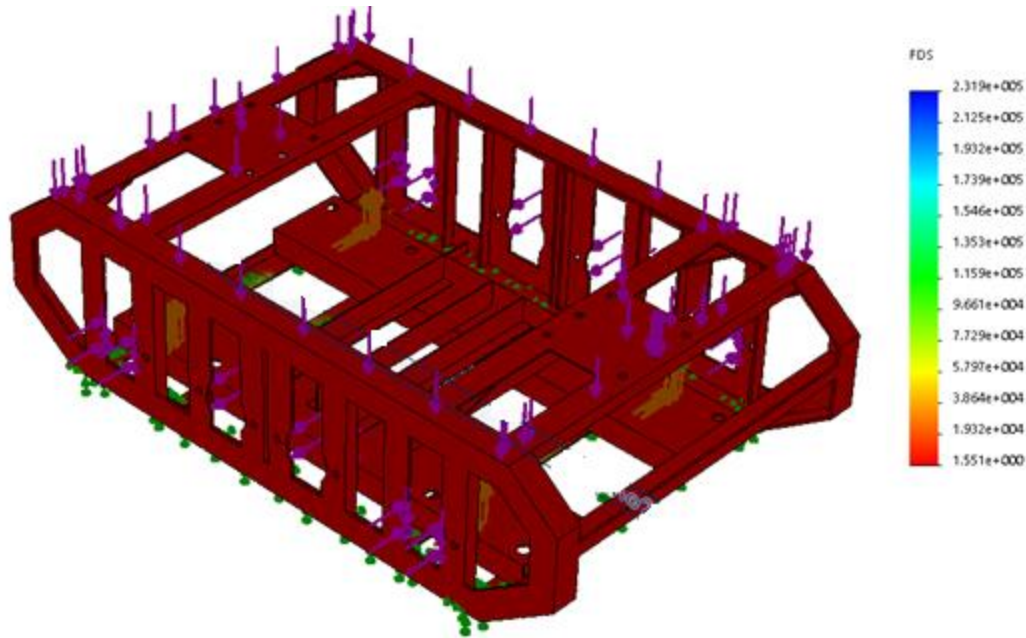


Figura 3.18 Aplicación de fuerzas para determinación de factores de seguridad.

El valor mínimo del factor de seguridad calculado se encuentra en el nodo 32189 y es de 1.55, valor que supera el límite de 1.5 para considerar al diseño como seguro.

3.1.5 Diseño del sistema electrónico y de control.

Luego del diseño y la construcción, respetando estos parámetros de diseño, de la plataforma robótica omnidireccional, se procede a hacer la selección (dependiendo de las características mecánicas vistas) de los elementos más adecuados que lleven a cabo el movimiento y operación coherente de todas estas piezas ensambladas y probadas, para lo cual se debe considerar los elementos electromecánicos que proporcionan la fuerza de desplazamiento de las ruedas y por ende de la plataforma completa, los sensores que en este caso son más influyentes que en el caso anterior, para determinar las posiciones del robot en tiempo real, los sistemas de comunicaciones que permitan establecer éste tiempo real, los controladores de la cinemática del robot, de los algoritmos de aplicación y etc.

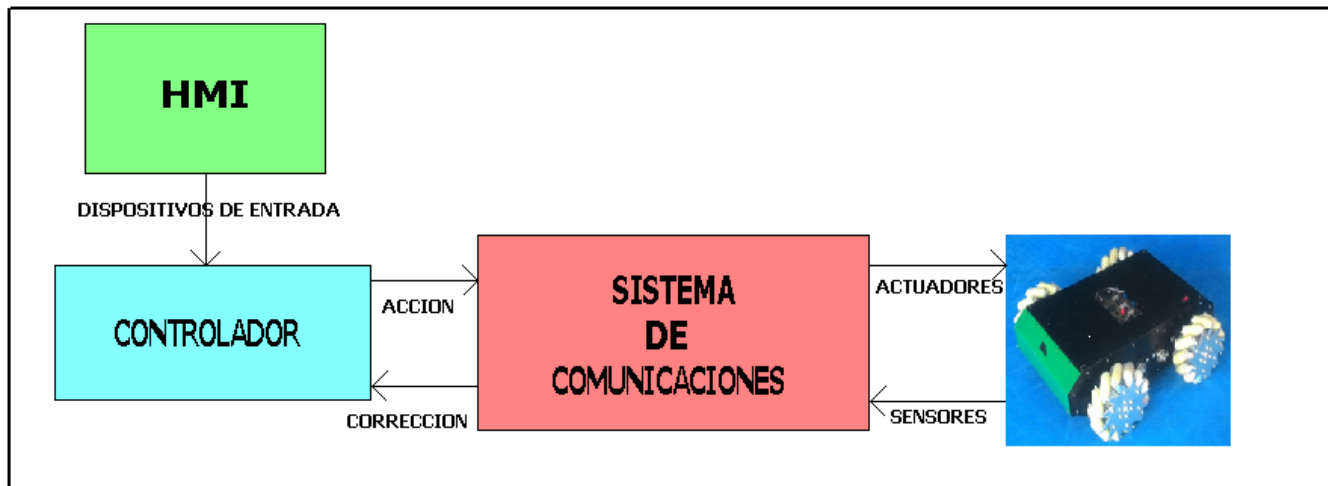


Figura 3.19 Diagrama de bloques del elemento electrónico de la plataforma omnidireccional.

En la Figura 3.19 se puede observar el esquema general del sistema electrónico y de control del robot omnidireccional, aquí se puede observar claramente los elementos que lo componen, no se ha definido en la gráfica el sistema de alimentación, que es muy similar al del robot uniclo construido, y que se encarga de proporcionar la alimentación a los elementos móviles y de control de la plataforma. Más adelante se describirá rápidamente su operación y sus diferencias fundamentales con el sistema del robot uniclo.

Los actuadores son cuatro motores eléctricos de DC, que impulsan el mecanismo de cadena – catalina para transmitir la fuerza del motor hacia las ruedas omnidireccionales, estos deben tener la suficiente potencia para poder arrastra al cuerpo de la plataforma y algunos elementos de aplicación particular más; Requieren una potencia de 560w, que, al distribuirla para las cuatro ruedas motrices, cada motor debe poseer una potencia de 140w. Los motores Dynamixel usados en la plataforma anterior son escogidos para formar parte de esta plataforma también, por todas las características mencionadas anteriormente y por la particularidad de ser motores de 200w cada uno, esto se debe a que son extremadamente compactos, su potencia es muy elevada debido a que cuentan con una caja reductora de velocidad razón por que se tiene una baja velocidad de operación, el voltaje de trabajo es mucho menor entre los otros candidatos, además algo muy importante que posee una tarjeta de control incorporada con sus propios sensores, circuitería de corrección de errores y una avanzada interfaz de comunicación industrial, esta circuitería incluye un controlador PID para corregir errores.

Para la determinación de la posición del robot en tiempo real, se necesita una medición primaria básica que es realizada mediante la técnica de odometría, pues se utiliza la información de la rotación de las ruedas y trasladar esta rotación a un sistema de vectores conveniente y así estimar la operación. Puede creerse que al ser ruedas omnidireccionales la odometría cambiaría su

concepto y los elementos de medición de la rotación tendrían que funcionar de forma distinta, pero más bien es el sistema de transformación de la información de rotación la que es diferente.

Para medir esta mencionada rotación de las ruedas, se utilizan Encoders ópticos en cada rueda, adicionalmente se incorpora mecánicamente un tacómetro al eje de rotación de las ruedas, de tal forma que se obtiene un sistema redundante de medición de la rotación de la rueda del robot, este tacómetro incorporado genera señales de voltaje proporcionales a la velocidad de giro de las ruedas.

Los sensores de temperatura son necesarios para la medición del calor generado dentro del motor, que es proporcional también a la corriente del motor, parámetro directo para saber la potencia que está desarrollando el motor, muchas veces es conveniente medir esta temperatura paralelamente a otros parámetros (si es que el motor nos posibilita) para saber cuándo puede encontrarse la operación cerca de una condición crítica, que incluso puede destruir el motor. Afortunadamente los motores Dynamixel, poseen incorporados sensores de temperatura que entregan sus valores por direcciones establecidas en su controlador, de esta forma solamente se cablea estas direcciones para obtener dicha información.

a. Elementos de control.

Si bien es fundamental la electrónica de movimiento y medición de la posición del robot en una aplicación, debe existir un control inteligente que los integre y opere en conjunto, es decir tome decisiones en base a la información leída y sea capaz de llevarla a cabo; estos elementos inteligentes, que además permitan desarrollar las aplicaciones dentro de ellos, son determinantes en nuestras plataformas, es por eso que se debe seleccionar el elemento más efectivo y adecuado a condiciones exigentes de nuestras aplicaciones.

Como se había explicado extensamente al inicio del capítulo, las alternativas de proceso son muy grandes, desde tarjetas de bajo costo y altas prestaciones, hasta computadores industriales dedicados, de decenas de miles de dólares; en nuestras plataformas fundamentalmente se han utilizado dos tecnologías, la primera y de un costo moderado, es el uso de computadoras a bordo de cada robot, con las ventajas y desventajas que ellos representa, rapidez de ejecución, recursos de programación ilimitados, amplia conectividad, gran capacidad de almacenamiento, etc. Y la segunda, la distribución de todo lo mencionado en elementos de bajo costo, es decir hacer que resida el control principal en una computadora remota y desde allí se distribuyan las señales de control a través de una red de comunicaciones y establecer la electrónica de control del movimiento en la propia plataforma con tarjetas de bajo costo y altas prestaciones.

Para este robot se utilizó un ordenador con norma industrial de robustez, impermeabilidad, etc. Se trata de un ordenador Eglobal mini PC i7, aprovechar las prestaciones inmensas de su

procesador y amplia periferia de comunicaciones y fundamentalmente por su requerimiento de alimentación de 12 Vdc, la posesión de dos antenas Wifi, 6 puertos USB, en una carcasa resistente y reducida. Una ventaja adicional es que, vista la necesidad de enviar los datos de la computadora central hacia los actuadores, se puede usar un conversor (USB2Dynamixel) directamente desde la PC a los actuadores, comúnmente esto se realiza usando señales moduladas desde una salida estándar o especial de la PC/tarjeta hacia los motores; en este caso se usa esta ventaja, que permite también gestionar los retardos eventuales y las decodificaciones o demodulaciones de la señal.

Este conversor que permite transformar la información proveniente de un computador a información legible para dispositivos de potencia ya sean motores, electro válvulas, etc. Lo realiza usando un bus de comunicaciones basado en RS485, base física de MODBUS; otra ventaja es la sencillez de su configuración y su entorno multiusuario y un depurador serial RS232 TTL.

b. Suministro de energía.

Para alimentación de todos los sistemas (igualmente que en el robot anterior) se utilizan batería LIPO por su versatilidad (pequeño tamaño, rápida descarga, ligeras, etc.), el dimensionamiento de las mismas se realizó con las tomando en consideración los consumos máximos de sus elementos:

$$P_T = P_{Computador} + P_{Motores}$$

$$P_{Motor} = V_{Motor} * I_{Motor}$$

$$P_{Motor} = 24V * 5A$$

$$P_{Motores} = 4 * P_{Motor}$$

$$P_{Motores} = 4 * 120 = 480watts$$

$$P_T = P_{Motores} + P_{pc}$$

$$P_T = 15watts + 480 watts = 495watts$$

Con estas consideraciones son necesarias 4 baterías de acuerdo a:

$$NumBat = \frac{P_T}{P_{Bat}}$$

$$P_{Bat} = V_{Bat} * I_{Bat} = 25,9V * 5,8 A = 150.22$$

$$NumBat = \frac{495 watts}{150.22 watts}$$

$$NumBat = 3,3 \approx 4 Baterías$$

La filosofía general del sistema de alimentación se mantiene, es decir el uso de tres placas fundamentales, la de potencia, que entrega la energía a los actuadores de los motores para su funcionamiento, la de control que integra la computadora industrial con sus elementos y periféricos y la de conexión adicional de la periferia descentralizada. En la Figura 3.20 se muestra la instalación de las placas y la computadora en el robot.

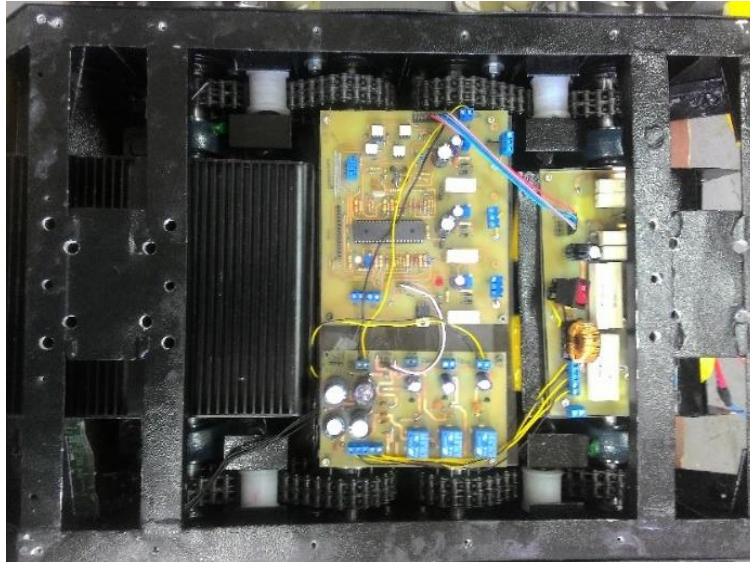


Figura 3.20 Montaje de la electrónica de potencia y control en el robot omnidireccional.

Como se puede observar, los dos robots descritos anteriormente, han empleado sistemas de muy alta calidad y altos costos, tales como los motores, actuadores, controladores y adaptadores Dynamixel, así como los ordenadores de control, y demás dispositivos, esto se pudo realizar únicamente por la participación en el proyecto CEDIA (Tele-Operación Bilateral Cooperativo de Múltiples Manipuladores Móviles) desde el año 2015 hasta la presente fecha, pues los altos costos que significaba invertir estos recursos para el desarrollo de la presente tesis imposibilitaba su desarrollo, pero mediante esta participación se pudo utilizar estos robots a más de unos de bajo costo contruidos personalmente, para las pruebas de las aplicaciones que permiten en esta tesis alcanzar todos nuestros objetivos y comprobarlos de forma totalmente práctica. Es así que además de estos robots como he mencionado, se construyeron también dos plataformas omnidireccionales más y una Uniciclo, que se las detalla a continuación, y que sirvieron eficientemente y que lograron definir más aun la conclusión principal de esta investigación, que se desarrollará de forma completa más adelante.

3.1.6 Construcción de la plataforma OMNIBEE.

Teniendo en cuenta la configuración del robot omnidireccional, los diferentes componentes como motores, ruedas mecanizadas, encoders y diferentes piezas se utilizan, que, en su conjunto ayudarán a la plataforma a moverse en todas las direcciones en el plano, la fuerza del robot omnidireccional para producir el movimiento viene dada por los cuatro 4 motores que se acoplan directamente a cada rueda omnidireccional del robot tipo Mecanum.

En el diseño de la estructura se tienen en cuenta algunas propiedades que el robot debe tener: es decir, el peso que soporta, la distribución de los elementos mecánicos y electrónicos y una geometría adecuada para su desplazamiento. En el diseño de la estructura se considera la fuerza y la estabilidad, por lo que la plataforma tiene una forma rectangular de un prisma, usando aluminio. En la Figura 3.21 se observa la simulación de las tensiones aplicadas a la estructura diseñada.

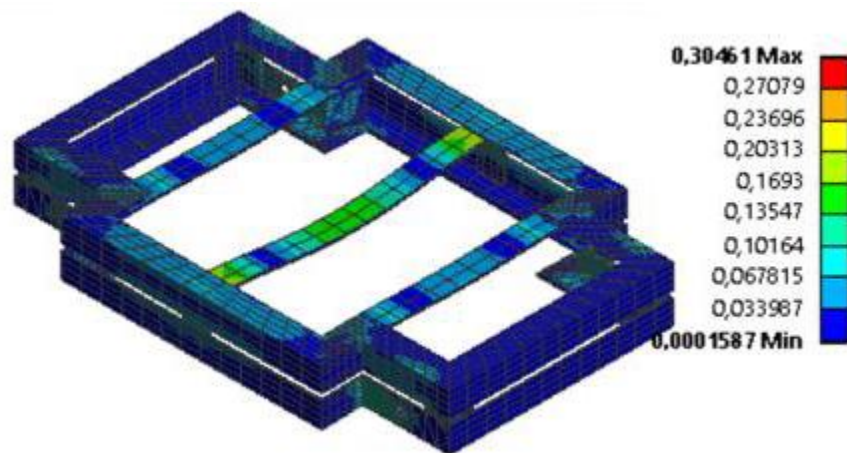


Figura 3.21 Simulación de las respuestas a las tensiones aplicadas a la estructura del robot OMNIBEE

Mediante la simulación, se determina el comportamiento de la estructura, que se somete a una fuerza real que se aplica en los lugares donde tiene la mayor carga, Figura 3.21. Mediante el análisis de elementos finitos se calculan las tensiones o deformaciones y presiones a las que está sometido, La Figura 3.22 indica el desplazamiento del material al aplicar una fuerza (color azul), se observa que las deformaciones causadas por las fuerzas aplicadas son muy pequeñas, por lo tanto, no se tiene en cuenta la rigidez causada por la carga. Este robot por sus características se empleará en aplicaciones en donde el movimiento y la velocidad explotan su versatilidad, ya que el peso total de la estructura construida nuevamente en aluminio y los componentes internos, son insignificantes.

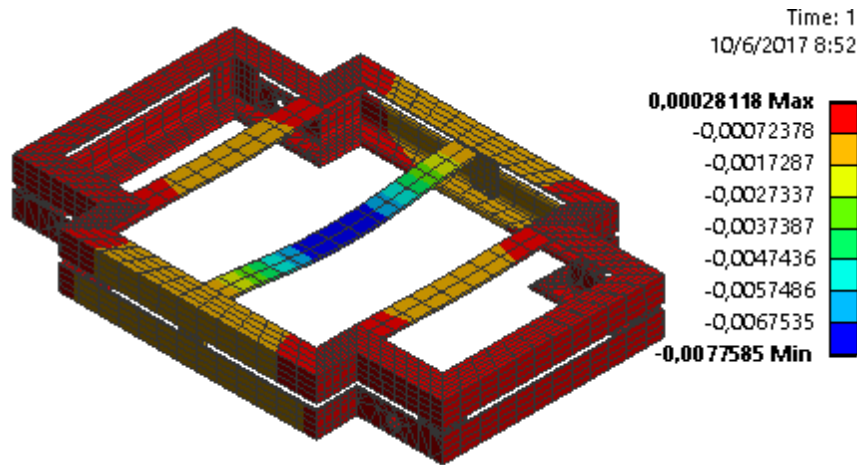


Figura 3.22 simulación del desplazamiento por tensión en la estructura OMNIBEE

a. Hardware e implementación de la interfase.

El hardware del robot omnidireccional se compone de cuatro etapas, es decir, consta de estructuras sub electrónicas que permiten al robot moverse dentro del plano X, Y y girar alrededor del eje Z. Las etapas que se han tenido en cuenta para la construcción del prototipo son: i) sistema de alimentación, ii) sistema de control, iii) etapa de actuadores y sensores, finalmente iv) la etapa de comunicación inalámbrica, en la Figura 3.23 se observa esto gráficamente.

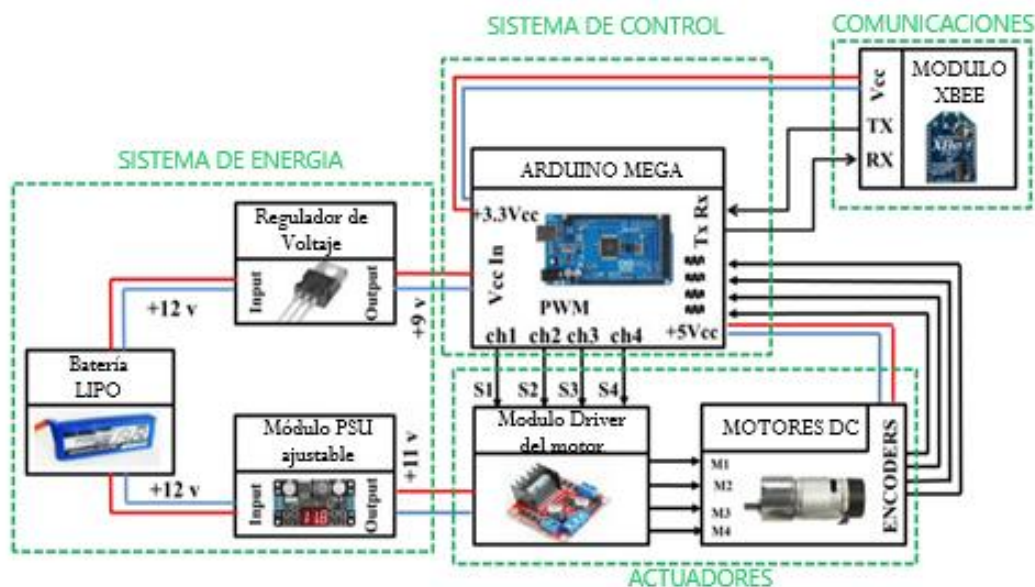


Figura 3.23 Diagrama de bloques de la plataforma OMNIBEE

El **Sistema de alimentación.** - del robot omnidireccional OMNIBEE consta de una sola batería LIPO (al contrario de las 4 de la plataforma anterior) de 4 celdas de carga con una intensidad de corriente de suministro de 2.2A, suficiente para proporcionar la corriente necesaria para alimentar a todo el sistema del robot; el voltaje de suministro para los drivers de los motores es de 11V, consta con un regulador de voltaje que ajusta al valor máximo aceptable por lo motores. Para el suministro de voltaje de la placa de control se empleó una fuente reguladora de 9V; la tarjeta de control a la vez entrega 5Vcc y 3.3Vcc para la alimentación de los sensores y para el módulo de comunicación respectivamente.

Sistema de control del movimiento del robot. – Está compuesto por una tarjeta Arduino Mega, que realiza varias tareas tales como: recibir las señales de las velocidades a configurar en cada una de las ruedas del robot, estas han sido obtenidas a partir del cálculo de las velocidades lineal y angular que se requieren mantenga el robot, en el seguimiento de una trayectoria, en una aplicación propuesta por un usuario; además realiza el control interno de los actuadores de forma que se pueda compensar la dinámica del robot pues los parámetros que operan en el mismo son la velocidad angular deseada W_d de cada rueda del robot y W_s representa la velocidad angular medida del robot, obtenida de los sensores, donde el error es compensado por el controlador interno incorporado en la placa que se muestra en la Figura 3.24 y está definido por:

$$W(t) = W_d(t) - W_s(t)$$

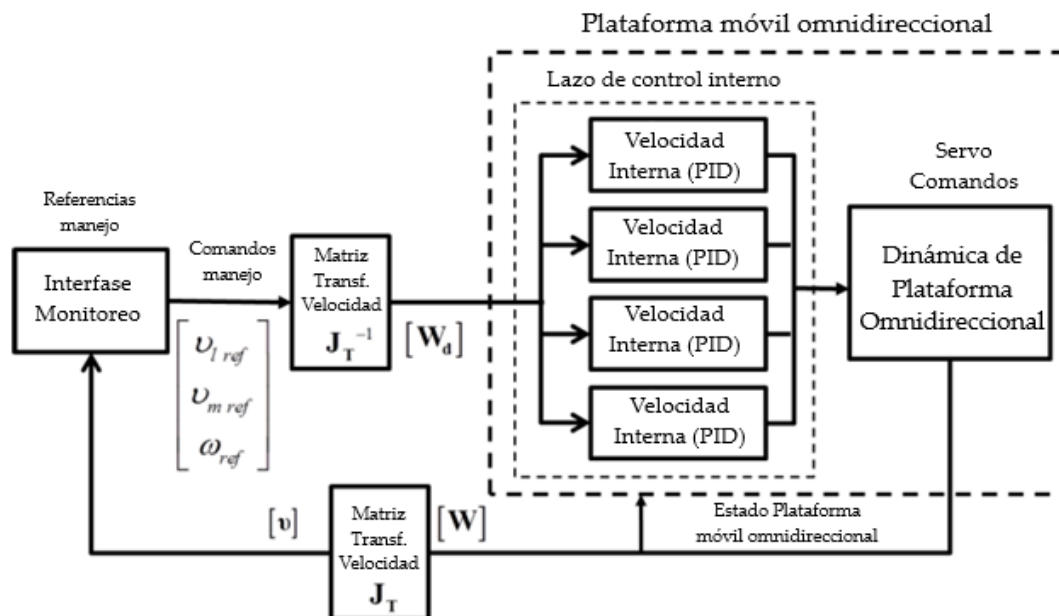


Figura 3.24 Diagrama de bloques del control de compensación del robot OMNIBEE

Como se puede observar y a partir de la experiencia del manejo de los motores Dynamixel en los robots construidos en las secciones anteriores, se abre el código de los actuadores para verificar la forma de control de compensación o control de error que estos realizan y se lo investiga, llegando a establecer que es una forma de control PID dentro de él, entonces se desarrolla un algoritmo basado en este tipo de control para la construcción e implementación del movimiento con todos los robots que se diseñen posteriormente.

La ecuación (3.2) define el controlador propuesto que se utiliza para compensar la dinámica del robot.

$$W_{dj}(t) = kp_j \tilde{W}_j(t) + ki_j \int_0^t \tilde{W}_j(t) dt + kd_j \frac{d}{dt} \tilde{W}_j(t) \quad (3.2)$$

Donde $\tilde{W}_j(t) = W_{dj}(t) - W_{sj}(t)$ con $j = 1,2,3,4$ representa el error de las velocidades angulares para cada actuador, mientras que kp_j, ki_j, kd_j son las constantes o ganancias positivas usadas para el control de los errores.

Como se puede visualizar entonces las velocidades angulares de las ruedas de los robots se corrigen con un paradigma de control realimentado, dando lugar a una reducción del error en la actuación de los motores y el movimiento entero del robot, este proceso es implementado en todas las plataformas construidas.

Los actuadores son motores de corriente continua de prestaciones generales, y dependiendo de su aplicación de mayor torque o velocidad o relación de transmisión, etc. En el caso del robot OMNIBEE consiste de motores de 11 voltios que contienen un encoder incorporado en el eje de cada motor, que permite medir los pulsos producidos y transformarlos con relaciones matemáticas adecuadas a sus medidas y características, a velocidades angulares para introducirlas a la tarea de control. El sistema general de los motores incluye un driver que controla la velocidad de cada motor o rueda mediante la recepción de una señal PWM y acciona de acuerdo a ella, cada motor.

Uno de los aspectos determinantes en el funcionamiento de este robot es la forma de control, ésta reside en un computador remoto, para hacerlo más flexible al momento de monitorearlo y supervisarlos y más versátil su control, pero al residir los principales algoritmos de control en un ordenador remoto, se necesita implementar una red de comunicaciones, que sobrepase los obstáculos de latencia y retardos que pueda la red presentar, de tal forma que estos no influyan de forma determinante en el comportamiento del control de movimientos del robot.

Obviamente se usa comunicaciones inalámbricas para generar las velocidades y transmitir las al control de movimiento, esto se hace en la plataforma OMNIBEE mediante una red inalámbrica ZigBee o el estándar IEEE 802.15.4.

b. Interface.

El control del robot, se ha basado en sus modelos cinemático y dinámico, y de acuerdo a su cálculo, se han establecido las relaciones matemáticas para desarrollar algoritmos que permitan mover remotamente la plataforma. Desarrollados los algoritmos a partir de los modelos y aplicaciones, se los programa en un lenguaje matemático que permita utilizar toda su potencialidad, lejos de escribir todas las funciones que son necesarias. La programación se la hace en software MATLAB y también se implementa una interfase humano máquina, con el fin de colocar los comandos de información necesaria para los movimientos del robot. MATLAB presenta la ventaja de realizar grandes y complicados cálculos matemáticos, además de permitir el desarrollo de una interfase, lo que aventaja más la implementación de este robot, pues de no ser así se tendría que realizar una conexión con otras aplicaciones, que tal vez puedan basarse en DDE, COM u otros. La interfase desarrollada permite operar automáticamente la plataforma se basa en dos modos de operación: modo automático y el modo de teleoperación.

En el *modo automático*: La comunicación se establece entre el sistema de control que genera las velocidades de referencia apropiadas para comandar autónomamente al robot, en el caso de estudio antes de establecer las aplicaciones, el sistema de control, que depende de la aplicación realmente, es quien genera las velocidades de referencia v_{ref} adecuadas para el movimiento del OMNIBEE y de esta forma lo gobiernan de forma automática, para el caso de estudio, estas velocidades podrán ser generadas por el operador de tal forma que se pueda validar la operación del prototipo. El sistema remoto de computación, recibe estas velocidades y a través de la implementación de la ecuación de control (4.10) se obtendrá las velocidades angulares deseadas para el robot W_d .

Las velocidades angulares medidas (por el escalamiento de la lectura de los encoders) serán transmitidas desde el robot al sistema informático, obteniendo de esta forma un lazo cerrado de todo el sistema, que permite ya implementar un control. Este control se desarrolla como había mencionado anteriormente en software MATLAB; mediante la Ec. (4.8) se determinan las velocidades lineales v_l y v_m y la velocidad angular ω medida en tiempo real. Para el caso de estudio de las pruebas y la confiabilidad de los resultados para emplear en las aplicaciones los valores enviados y recibidos entre el robot y la computadora son graficados y contrastados.

La Figura 3.25 indica la interfase de operación en modo automático.

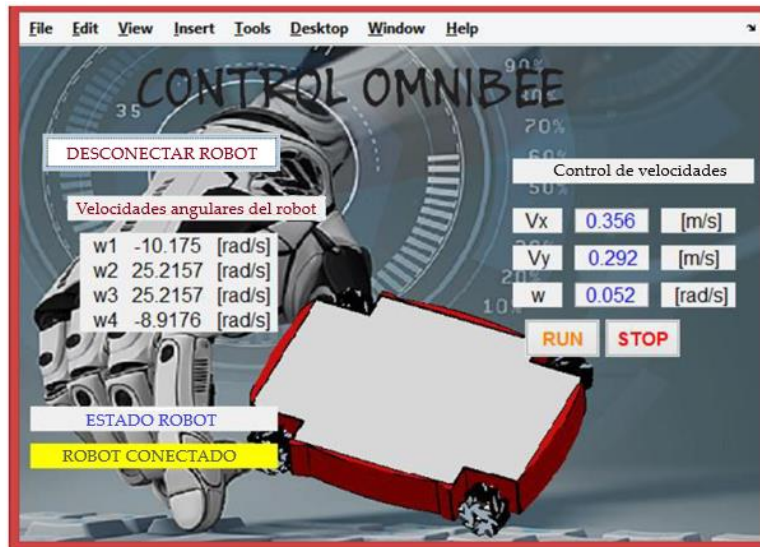


Figura 3.25 Interfase de operación automática del OMNIBEE

En el *modo de Teleoperación* el robot es completamente controlado por un usuario, quien genera los comandos de movimiento presionando los botones del HMI desarrollado, el cual puede utilizarse para realizar en línea y en tiempo real, movimientos lineales verticales, horizontales, hacia atrás, a la derecha, izquierda, etc.; puede rotar en su propio eje, y describir movimientos diagonales a este o desplazados 45 grados. La interfase envía los comandos actualizando las velocidades para cada motor, el robot produce los movimientos esperados a una velocidad constante definida por el operador. En la Figura 3.26 se observa la interfase desarrollada para este procedimiento.



Figura 3.26 Interfase de tele operación del OMNIBEE

3.1.7 Construcción del Manipulador móvil.

Dentro de las metas y actividades planteadas en la presente tesis, estaba el estudio del consumo de energía y el uso de técnicas para la reducción y gestión de la energía en robots móviles.

Para no alargar mucho esta explicación, el uso de las baterías LIPO en todas nuestras plataformas es una forma determinante de gestión de la energía entregada a un robot, porque si bien es cierto, de algún modo estas baterías son inestables de acuerdo a su carga/descarga y otros aspectos, bien administradas, es decir estableciendo cronológicamente el cumplimiento de los períodos de carga y descarga (en períodos de desuso) y los límites de corriente de carga, etc., estas baterías son lo mejor que tienen actualmente el mercado, duran más en tiempo de operación, rápidas en tiempo de carga y descarga, etc.

Entonces, este estudio para la reducción de energía en la operación de los robots móviles se centraba en dos ejes fundamentales: 1) el consumo de energía de los elementos eléctricos y electrónicos que componen la plataforma y 2) el desarrollo de técnicas que permitan maximizar dichos consumos en la consecución de tareas.

El primer aspecto también puede dividirse en dos grandes líneas: 1) el consumo de los elementos netamente eléctricos, es decir los motores y 2) el consumo de los elementos electrónicos, es decir las placas de control, computadoras, sistemas de comunicaciones, etc. Tomando en cuenta este aspecto la verdad es que la operación en primer lugar de los motores eléctricos de un robot móvil, es continua y permanente y el consumo de energía depende mucho de la velocidad y potencia que se aplique en las tareas, y existe poca posibilidad que de alguna forma física se logre disminuir la energía que necesitan, por otro lado, las tarjetas y demás elementos electrónicos, tienen un consumo de potencia estándar en su operación, que también tiene que ser permanente, pero que si se puede operar de forma discontinua, y es lo que se ha hecho, utilizar los elementos electrónicos únicamente cuando son necesarios, pues están conectados a las placas pero mientras no operen, no consumen energía adicional, que si lo hacen cuando empiezan a operar, es decir se realiza una especie de multiplexación usándoles a los dispositivos solamente en etapas de los programas. La verdad es que esta técnica no reduce de forma significativa el consumo de energía de las baterías.

Entonces la alternativa fue definitivamente la de descubrir técnicas o estrategias que permitan a los robots móviles ahorrar energía en la ejecución de sus tareas, es decir en sus aplicaciones, pues en el movimiento, un robot móvil usa la energía que necesita para moverse y nada más. Es así que, para comprobar lo dicho, y demostrar que se puede conseguir este objetivo, se desarrollaron algoritmos que proyectaban ya la reducción del consumo de energía en tareas de manipuladores móviles, se diseñaron e instalaron sobre los robots construidos y probó que realmente funciona, es decir se demostró que sobre las plataformas móviles se puede reducir el

consumo en tareas de manipuladores móviles. Esto se mostrará más adelante; por ahora nos centraremos en la construcción del manipulador móvil.

a. Los brazos robóticos. - Están presentes actualmente en varias instancias de la vida del ser humano, abarcando grandes campos de aplicación desde la industria hasta la medicina. Los brazos robóticos emulan en gran porcentaje la operación de un brazo humano, pues no es posible emularlo por completo, por su complejidad y versatilidad; al igual que este tiene partes diferenciadas, como: hombro, codo muñeca, articulaciones que se desarrollan con mecanismos para unir eslabones o partes fijas, controlados con un sistema sumamente inteligente que proporcione esta emulación. Es de gran importancia en la industria no solamente por su eficacia sino también porque es capaz de sustituir en su función a un ser humano, con mayores prestaciones y mayor precisión. Los brazos robóticos suelen ser contruidos con diferentes configuraciones, entre ellas: cartesiana, cilíndrica, polar, scara, articulada, paralela, antropomórfica, etc.

b. Manipulador móvil. - Estos sistemas armonizan las ventajas de las plataformas móviles y los brazos robóticos manipuladores, por lo cual mejoran el rango de capacidad de trabajo dentro de este tipo de robots ya que su funcionamiento combina una plataforma móvil que amplía el área de trabajo, mientras que el brazo opera normalmente como en cualquier base uniforme. Esto hace que disminuya sus desventajas ya que un brazo normalmente se ve limitado por el área de trabajo. Los grados de libertad extra son otro punto a favor de la plataforma móvil entregando al usuario más opciones y más destreza en sus movimientos. Sin embargo, el funcionamiento y estabilidad de este sistema se torna un poco más complejo debido a los muchos grados de libertad y al medio de trabajo no estructurado.

Es un sistema que comprende un robot móvil y un brazo robótico, unidos mecánicamente, es decir una plataforma que se encarga del desplazamiento y un brazo que se encarga de la manipulación, la unión de estos sistemas hace que la versatilidad del elemento mejore las características desventajosas de cada uno, es así que el manipulador móvil puede hacer que la aplicación de una plataforma móvil llegue a sitios en donde por sí sola no lo puede hacer y el robot móvil hace que el campo de trabajo del manipulador no esté limitado a su área de trabajo.

El manipulador móvil está formado generalmente por tres sistemas: la plataforma móvil, el brazo robótico y el efector terminal; estas plataformas conjuntas también pueden tener diversas configuraciones como: manipulador humanoide móvil, brazo manipulador móvil, y manipulador móvil omnidireccional. En base a lo expuesto anteriormente se pueden construir los dos últimos tipos de manipulador móvil.

c. Construcción del Manipulador móvil.

Para la construcción del manipulador móvil, orientado a las aplicaciones más importantes que se pueden desarrollar, entre ellas, la demostración de los algoritmos de disminución de consumo de energía en las tareas asignadas a manipuladores móviles, se consideraron los aspectos más importantes que son:

1. Se elige la **configuración antropomórfica** ya que permite tener mejor orientación en su extremo operativo y puede moverse con mucha facilidad en trayectorias un poco complejas, y principalmente por su movimiento rotacional y varios angulares.
2. La **capacidad de carga** se proyecta para aplicaciones de no más de 0,5 kg.
3. Se considera que, para el posicionamiento y selección de un objeto, son suficientes seis **grados de libertad**: tres para establecer la posición y tres para la orientación.
4. Este parámetro está muy relacionado con las plataformas móviles ya construidas y las aplicaciones que con ellos se pueden implementar, en este sentido la **velocidad** se establece aproximadamente en 0.8 m/s pues las tareas y aplicaciones requieren cierta rapidez y mucha precisión.
5. Se considera que el **alcance máximo** ha de ser de 75 cm desde su base.

d. Geometría de la estructura.

Para la estructura del brazo se han considerados una serie de eslabones unidos mediante articulaciones de giro; se compone desde su base, de un tronco que es la unión con la base, un hombro unido con este tronco, un codo unido con el hombro, una muñeca unida al codo y la mano que es la que presentara la presencia del efector final; para ellos se utilizan diversos componentes como motores, como elementos eléctricos, eslabones, frames, articulaciones y el extremo operativo que servirá para las aplicaciones de sujeción de objetos. La Figura 3.27 muestra la configuración del brazo.

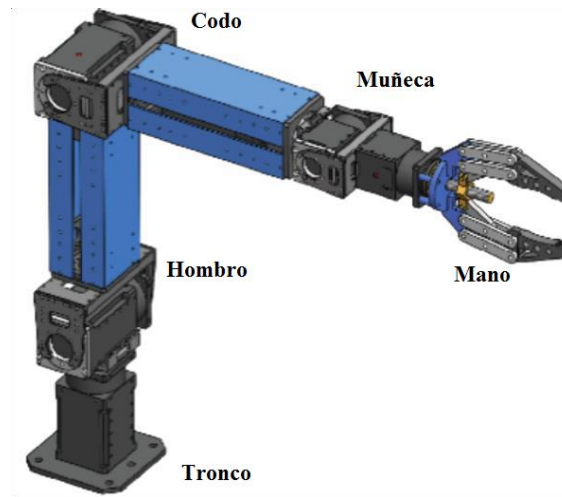


Figura 3.27 Configuración del brazo robótico del manipulador móvil

Con el fin de diseñar la estructura del brazo que se acoplará a la plataforma móvil es necesario se consideren las cargas que se van a aplicar a cada uno de los elementos, esto se calcula a través de los siguientes procedimientos:

1. Motor axial de la muñeca. Se realizan los cálculos requeridos para cada actuador, aplicando el principio fundamental de torque: $T = F * d$. Considerando la distancia que separa a cada una de las articulaciones se analiza como:

Para la muñeca:

ω_{mg} : peso motor gripper

d_{mg} : longitud motor gripper

ω_{ac} : peso del acople del motor

d_{ac} : distancia del acople del motor

ω_{gr} : peso gripper

d_{pb} : distancia centro del gripper

ω_{cu} : peso carga útil

P_b : punto de posicionamiento

T_{m1} : torque motor uno

En la Figura 3.28 se muestra la configuración de la pinza.

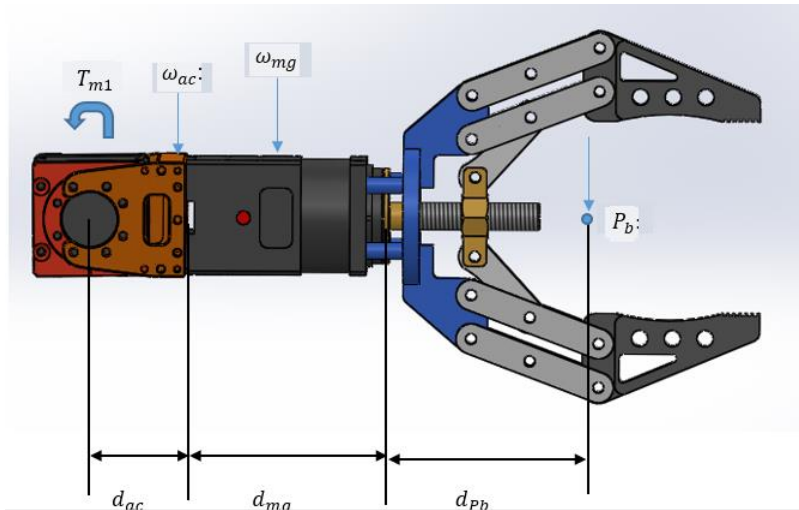


Figura 3.28 Articulación de la muñeca y la pinza

$$T_{m1} = \omega_{ac} \left(\frac{d_{ac}}{2} \right) + \omega_{mg} \left(d_{ac} + \frac{d_{mg}}{2} \right) + \omega_{gr} (d_{ac} + d_{mg} + d_{pb}) + \omega_{cu} (d_{ac} + d_{mg} + d_{pb})$$

$$T_{m1} = 0,6N \left(\frac{0,035}{2} \right) m + 2,55N \left(0,035 + \frac{0,072}{2} \right) m + 2,25N(0,035 + 0,072 + 0,069)m + 500N(0,035 + 0,072 + 0,069)m$$

$$T_{m1} = 1,48 Nm$$

Velocidad:

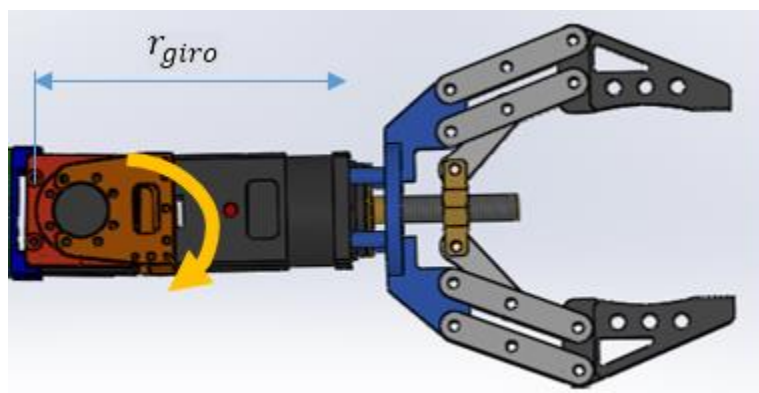


Figura 3.29 Giro de la muñeca

$$v = \omega \cdot r_{giro}$$

$$\omega = v/r_{giro}$$

$$\omega = \frac{0,5m/s}{0,13_m}$$

$$\omega = 3,8\pi rad/s = 12rad/s = 36rpm$$

Potencia:

$$P = T\omega$$

$$P = 1,48Nm(12rad/s)$$

$$P = 17,8W$$

Utilizando este mismo criterio, y cambiando los valores de los torques, las distancias de operación, las velocidades de cada segmento descrito a partir de la Figura 3.27, se calculan los otros parámetros (es decir la potencia necesaria en los motores) para cada motor de cada elemento móvil de acuerdo a la Figura 3.30. Obteniéndose los resultados mostrados en la Tabla 3.1.

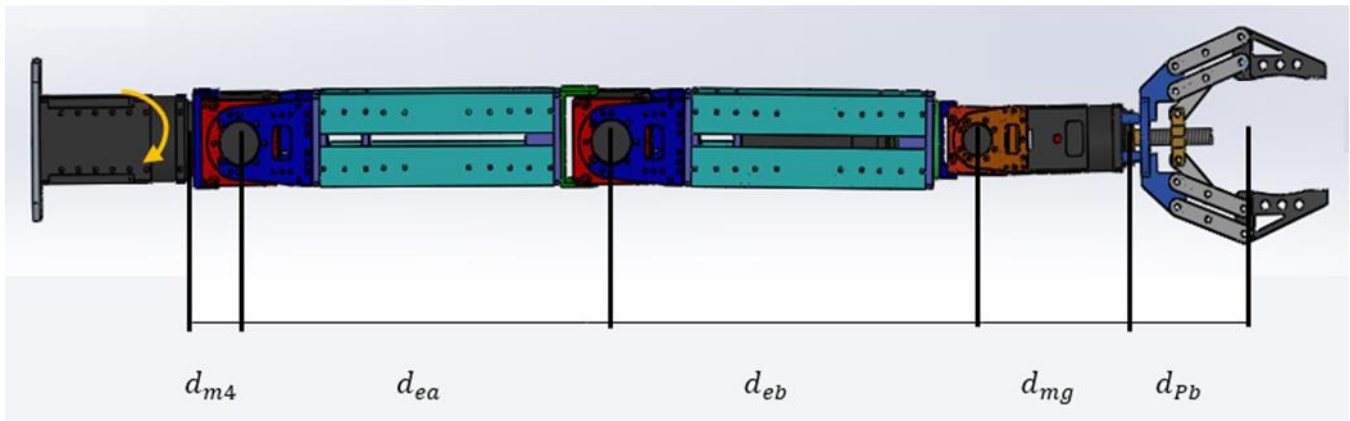


Figura 3.30 Elementos móviles del brazo robótico

Tabla 3.1. Parámetros constantes de los elementos móviles del brazo robótico necesarios

Parámetro	Torque	Velocidad	Potencia
Motor axial de la muñeca	1.48 Nm	12 rad/s	17.8 w
Motor de rotación de la muñeca	2.33 Nm	8.92 rad/s	20.8 w
Motor en la articulación del codo	6.27 Nm	3.92 rad/s	24.62 w
Motor en la articulación del hombro	14.6 Nm	3.6 rad/s	52.5 w
Motor en la articulación de la base	14.7 Nm	2.85 rad/s	42 w

Finalmente para calcular la potencia del motor que se necesita para establecer la fuerza de agarre o sujeción que tiene la pinza o gripper, se realiza un estudio de la fuerza que se transmite a partir de la rotación del motor hacia las dos tenazas de la pinza, que son los que se abren o cierran para sujetar un objeto; este sistema se lo ha realizado a través del sistema de tornillo de potencia, que consiste en un mecanismo que transmite potencia a partir de la rotación del tornillo (eje de un motor), de forma como se indica en la Figura 3.31. El resultado del cálculo es de 13,4 w.

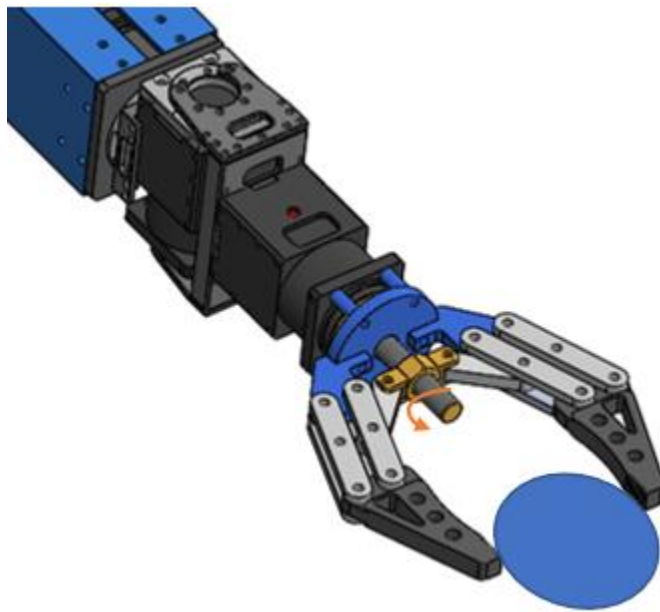


Figura 3.31 Mecanismo de tornillo de potencia.

De acuerdo con esto, la selección de los actuadores, obviamente eléctricos, es definida, además por sus parámetros máximos de potencia, también por su característica de auto control, velocidad, medida de su posición, y frenado, se han escogido nuevamente los motores de la marca Dynamixel de las series L54 y L42 en sus versiones 50, 30 y 10; esto porque han presentado respuestas a las exigencias de control superiores a los otros que se han probado (TowerPro y Hitec).

Finalmente se ha de destacar que el mecanismo utilizado para la transmisión del motor a las articulaciones es el accionamiento directo, en donde el eje del actuador se conecta directamente al eje de la articulación, sin uso de acopladores, reductores, bandas, etc., además de que, al ser actuadores de torque elevado, es fácilmente acoplable a este sistema de transmisión directo, que hace además que se incremente su velocidad de transferencia y su precisión.

En este robot (brazo) no existen placas electrónicas necesarias ya que como se mencionó antes los ejes de los motores se acoplan directamente a las articulaciones o seas a los ejes de movimiento de cada sección, y lo hacen únicamente (en lo eléctrico) con cables de conexión a

sus terminales, los cables se conducen por el interno del brazo y se han diseñado de tal forma que los movimientos no estorben a su conexión, ni que estas hagan que de alguna forma se traben o atasque en ninguna posición.

Como en los casos anteriores, con respecto a las estructuras, se realizan simulaciones de operación y carga para determinar su comportamiento, y determinar sus respuestas, especialmente de las articulaciones; la Figura 3.32 muestra los resultados de la simulación de los niveles de deformación del material, se observa en un rango de colores del rojo al azul respectivamente, la deformación del material, siendo el rojo el punto más crítico, en donde se concentrará el mayor esfuerzo.

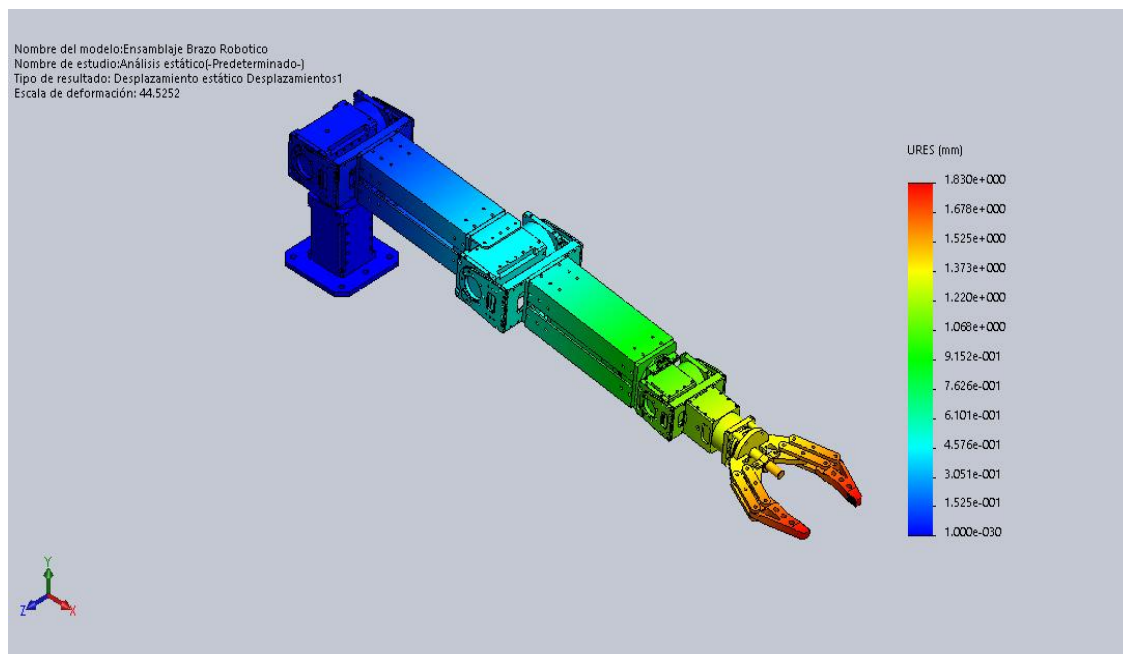


Figura 3.32 Análisis de los esfuerzos del brazo robótico.

El factor de seguridad (considerado también en las dos estructuras anteriores) es en este caso también mayor que uno, lo que significa que la estructura superará el valor máximo de la carga esperada, entonces se escogen los materiales con los que ha de construirse el brazo que son: perfiles de aluminio 6061T6, que es aluminio endurecido por una aleación de magnesio y silicio, que presenta las características necesarias suficientes para la operación del brazo.

e. Acoplamiento en la plataforma móvil.

Como se ha mencionado en las secciones anteriores, para las pruebas de las aplicaciones planteadas, que nos permitan cumplir con los objetivos propuestos en esta tesis, se han construido cuatro plataformas robóticas móviles, que serán empleadas en dichas aplicaciones. Uno de los objetivos del trabajo es la propuesta de un sistema que reduzca el consumo de energía

de los robots móviles en tareas de aplicación, este algoritmo, que será más tarde presentado, desarrolla técnicas de ahorro de energía en tareas asignadas a manipuladores móviles, es por eso que, es necesario adaptar el brazo robótico a una plataforma móvil; para ello podríamos hacerlo solamente con las plataformas construidas en las estructuras con geometría hexagonal construidas con estructura de aluminio, puesto que el peso del brazo no podría ser soportado por las otras dos plataformas, es así que se podría Montar (y así están diseñados) sobre las estructuras de primero y segundo robot presentados (no han sido nombrados) aprovechando este particular para definir también la ventaja de cada plataforma sobre la aplicación incluso. Entonces se pueden construir manipuladores móviles sobre robots unicycle o sobre robots omnidireccionales.

Se ensambló el brazo robótico para las pruebas del algoritmo de reducción de energía de las baterías, en la construcción del manipulador móvil en la plataforma unicycle, existe también la posibilidad en estos dos diseños, de montar otro brazo robótico en ellas, la Figura 3.33 muestra el montaje final del manipulador móvil. Se insiste, lo mismo se puede realizar en el robot omnidireccional también, la estructura permite la colocación de uno o dos brazos robóticos sobre él.

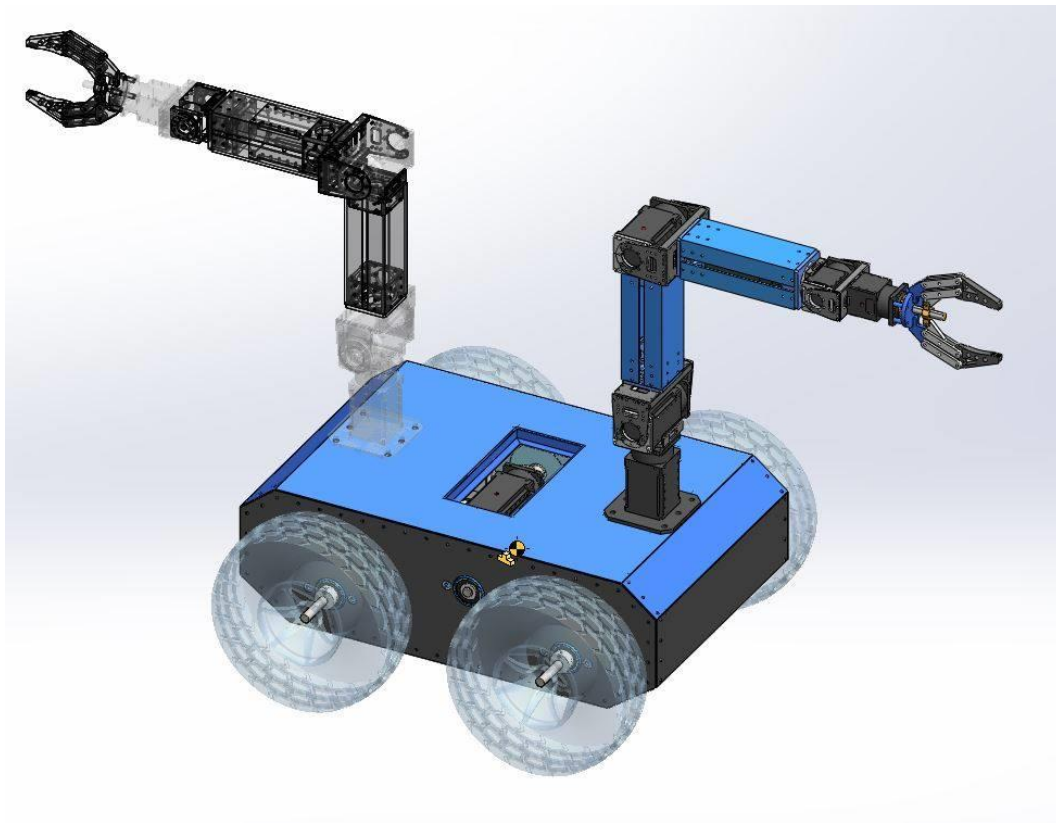


Figura 3.33 Manipulador móvil montado sobre robot omnidireccional con posibilidad de conectar un segundo brazo en su parte posterior.

Capítulo IV

4 MODELACIÓN MATEMÁTICA DE ROBOTS MÓVILES Y MANIPULADORES MÓVILES.

4.1 Introducción.

Los grandes avances en la electrónica contemporáneos, son el resultado de investigaciones requeridas por la creciente demanda de sus aplicaciones y ventajas en procesos de diversa índole, sistemas en los que, hasta hace pocos años se habría creído imposible la intromisión de esta ciencia y tecnología, son hoy en día aplicaciones solucionadas de formas increíbles, p.e., la electrónica de control dentro de un automóvil; es por esta razón que en muchos casos, las investigaciones se orientan más a la implementación de sensores para cualquier variable imaginada y sistemas que controlen procesos que involucren dichas variables, obviamente el primer paso es medir la variable de interés, sea la que sea, de forma electrónica; posteriormente se desarrollará un algoritmo que controle el proceso que ha medido esa (s) variable (s); ese es el objetivo.

Obviamente no se puede suponer únicamente, que la inventiva de los seres humanos se ha desarrollado desde la era de la electricidad y la electrónica, no es así, mucho antes de estos ya existían elementos mecánicos y electromecánicos que permitían de alguna forma realizar el control automático de un proceso; las ventajas de hacerlo en forma eléctrica, son las mismas que las de su naturaleza: fácil generación, transporte, distribución, libre de contaminación, etc.

Los sistemas de realimentación con el fin de controlar procesos tienen ya una historia recorrida, en la que se incluyen elementos puramente mecánicos, electro mecánicos y electrónicos; es así que por ejemplo el sistema de flotador para controlar el nivel de aceite de una lámpara ya era usado en Alejandría antes de nuestra era; y que históricamente pudiésemos citar los trabajos de James Watt en 1796, Iván Polzunov en 1765, etc. Como trabajos relevantes y pioneros en los sistemas de control automático, estos en el transcurso de muchos años aunaban esfuerzos para aumentar la exactitud del sistema de control, logrando reducir oscilaciones incluso en sistemas inestables.

Esto hizo que fuera imperativo formular una teoría matemática que condujera al uso eficiente de estos sistemas y tenga un respaldo completamente científico. J.C. Maxwell formuló una teoría matemática relacionada a la teoría de control, usando un modelo de ecuación diferencial de un regulador, en donde se consideraban los efectos que tenían los diversos parámetros naturales y mecánicos, dentro del comportamiento de dicho regulador. La época de la segunda guerra mundial, hace que se necesiten más sistemas de este tipo en el desarrollo del conflicto para la obtención de la supremacía y entonces se diseñan sistemas de control automático de tiro,

sistemas de pilotos automáticos para aviones, etc.; es por esta razón que estas teorías se desarrollaron rápidamente, vista la complejidad y exactitud de los sistemas militares, por lo que se aumentó el interés y desarrollaron nuevos métodos e ideas, basados en métodos matemáticos y analíticos. Al terminar la gran guerra, se centran más los estudios en el uso de la transformada de Laplace y los lugares geométricos de las raíces, incluyendo posteriormente ordenadores digitales en estos sistemas.

En la industria, los sistemas de control requieren necesariamente que no solamente se conozca a precisión una máquina, sino todos los procesos alrededor de ella, para que el proceso sobre el cual actúe un sistema de control sea eficiente y aumente la productividad. Por esta razón se habla del modelado de la planta y no solo del modelado de la variable, y el equipo que actúe sobre ella, porque sobre ellas también influyen agentes internos y externos que pueden hacerle cambiar su comportamiento y poner fin al control, o al menos desestabilizarlo.

La aplicación de un algoritmo de control sobre un sistema, requiere actuar sobre cada aspecto que pueda de alguna forma controlar cada uno de estos parámetros y prevenir la presencia de factores desconocidos, es por eso que el modelo de la planta debe acercarse lo más posible a la realidad de todos los factores involucrados; pero esto hace también el modelo más complicado, pues por cada elemento a controlar incluye en su forma matemática una variable, que muchas veces dificulta su proposición, pero que determina también la importancia sobre los factores que realmente se desean controlar y de qué forma; independientemente de esto, de esta forma se determina la gran importancia de establecer el modelo matemático de la planta para poder llevar a cabo de la manera más eficiente su operación, disminuyendo sus errores y además prevenir hechos poco evidentes, que pueden llegar a causar desde errores en el control y hasta destrucción de los elementos incorporados.

Los robots móviles, brazos robóticos y manipuladores móviles, no son la excepción, al contrario, son sistemas que, dependiendo de su aplicación, no toleran en absoluto errores, p.e.: los robots empleados en medicina y en interacción con pacientes humanos graves. Es por esta razón, lo determinante de su modelación matemática antes de implementar un algoritmo de control y más aún cuando este algoritmo le permite corregir los errores e implementar sistemas que mejoren su estabilidad. Pero obviamente, no es lo mismo controlar un robot móvil de tipo Car-like, un unicycle o un omnidireccional, etc., además de que su tamaño (todas sus dimensiones), su peso, estructura, el tamaño y peso de sus componentes, etc., también determinan parámetros que influyen en la “carga” del sistema de control al momento de diseñarlo y más aún al implementarlo y ponerlo bajo prueba.

La modelación matemática de un robot móvil se vuelve entonces de suma importancia, pues mediante ella, se implementarán de forma confiable los algoritmos de aplicación; para el caso

de esta tesis: a) algoritmos de colaboración, b) algoritmos que permiten la reducción del consumo de energía, y c) algoritmos de búsqueda y localización.

Un robot móvil, por su concepción no deja de ser una “planta” que se debe modelar para establecer el control de sus características en una aplicación, es así que el modo de control realimentado, Figura 4.1, se emplea también sobre ellos, así como sus definiciones: Variable controlada, es generalmente el movimiento angular y lineal que los robots suelen describir; Planta, es el robot en sí mismo con todas sus características y afectaciones; Procesos, son las operaciones que realiza en una aplicación, comunes o excepcionales; Sistema, es el conjunto de todos los componentes incluidos los que no están montados sobre el robot, como las comunicaciones, el ordenador, etc.; y las perturbaciones, todo lo que afecta al robot en su tarea de moverse en forma lineal y angular, de ahí, que existan otro tipo de perturbaciones al momento de realizar las tareas en sus aplicaciones deberán modelarse en el control de la aplicación; finalmente el control realimentado, también se refiere a la operación de corrección sobre los parámetros del robot cuando se presenta una de las perturbaciones, con el fin de corregir los errores que esta introduzca en su tarea.

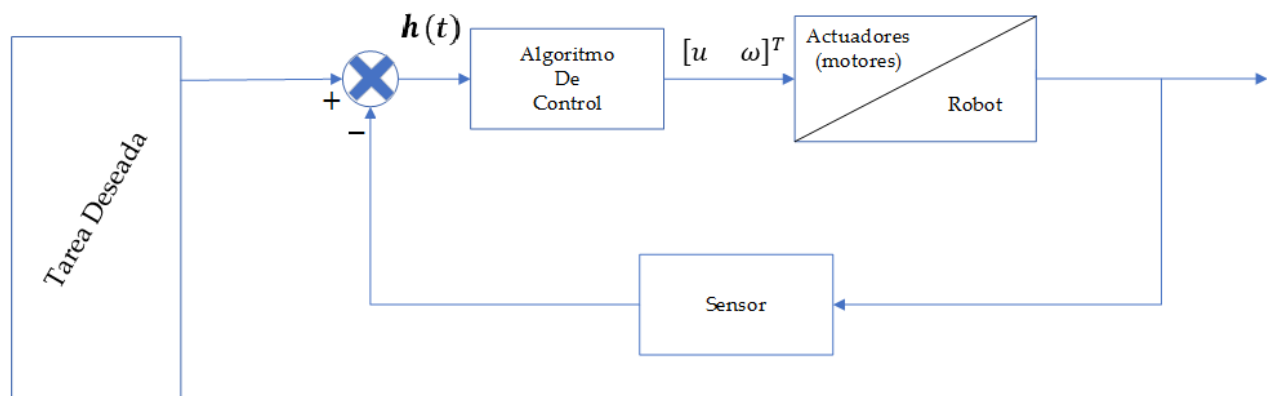


Figura 4.1. Lazo de control realimentado para robots móviles.

4.2 Modelos matemáticos de sistemas de control de robots móviles.

Cuando se consideran sistemas de control para robots móviles, se debe tener en cuenta que una de las tareas más importantes en el análisis y diseño del sistema es establecer de forma precisa su modelación matemática, esto se puede hacer por el uso de funciones de transferencia o variables de estado, según el tipo de control que se desea aplicar, que depende directamente de las aplicaciones que se requieran, es decir si presentan o no características invariantes en el tiempo y también su carácter lineal o no. Esto requiere necesariamente de no solamente saber

describir el sistema de forma matemática sino también de ser capaz de hacer suposiciones y aproximaciones correctas para caracterizar el sistema de su forma más real, incluso cuando actúa sobre su ambiente más extremo, se puede hacer uso también en este empeño, de sistemas de simulación computarizados.

Si bien es cierto que un robot móvil es un sistema que contiene muchos elementos eléctricos y electrónicos, y muchos de ellos altamente sofisticados, por ejemplo aquellos que introducen inteligencia artificial y sistemas expertos en sus tareas de operación; no es menos cierto que también contienen muchos elementos mecánicos que considerar, ya que ellos también se incorporan a la modelación matemática de los robots; es por esto que en la mayoría de los robots, se ha empleado técnicas de diseño mecánico también, para facilitar esa determinación de suposiciones, caracterizaciones o aproximaciones que se necesitan en la modelación y se recogen en los resultados, especialmente porque las aplicaciones tienden a incluir movimientos lineales y angulares.

En un robot se deben siempre considerar dos tipos de modelos: el modelo cinemático que ilustra el movimiento del robot y el modelo dinámico que determina las relaciones entre las entradas y las salidas considerando: fuerzas, masas, rozamientos, fricciones, etc.

4.2.1 Modelo Cinemático de los robots móviles, generalidades.

Los primeros robots en realizar aplicaciones industriales y relacionadas fueron los robots manipuladores también conocidos como brazos robóticos, en los cuales se analiza la disposición de sus articulaciones para definir el modelo que se hace a través de una cadena cinemática abierta; la principal diferencia con un robot móvil es que se considera como un conjunto de cadenas cinemáticas cerradas cuyo número está determinado por la cantidad de ruedas motrices y algunas veces por las ruedas que estén en contacto con el suelo; normalmente se encuentran las siguientes restricciones al desarrollar el modelo cinemático de un robot móvil:

- Que el robot se desplace por una superficie plana,
- Que no existan elementos flexibles en la estructura del robot,
- Que los ejes directrices del robot sean perpendiculares al suelo, y
- Se desprecia todo tipo de fricción en el movimiento del robot.

Un aspecto muy importante al considerar el modelo a establecer, es el de sus características holonómicas o no holonómicas, ya que, se puede constituir un sistema de ecuaciones con variables de velocidad que logren integrarse, y de esta forma conducir el sistema de restricciones a uno con variables de posición.

4.2.2 *Modelo dinámico de los robots móviles, generalidades.*

Como se dijo en los párrafos anteriores, el modelo cinemático del robot evidencia e ilustra los movimientos que se pretende aplicar en él; en cambio los modelos dinámicos, dan la posibilidad de determinar las relaciones entre las entradas y las salidas consideradas, para efectuar dichas operaciones, esto considera: las masas relacionadas, las fuerzas, etc. Debe, y es muy importante conocer las características holonómicas del agente para establecer adecuadamente su modelo dinámico también.

Los modelos dinámicos se pueden obtener de muchas formas matemáticas firmes, como los métodos de Newton-Euler, Euler-Lagrange, el método generalizado de D'Alembert y Lagrange, Euler y Hamilton, etc.; sin embargo, hay dos métodos principales para determinar las ecuaciones dinámicas en este tipo de sistemas mecánicos: las ecuaciones de Newton – Euler basadas en las tres leyes de Newton y; las ecuaciones de Euler – Lagrange de cálculo variacional.

La diferencia principal entre los dos enfoques está relacionada con las restricciones. Mientras que las ecuaciones Newton tratan cada cuerpo rígido separadamente y modela explícitamente las restricciones a través de las fuerzas requeridas para dominarlas, El método de Lagrange y D'Alembert ofrece los procedimientos sistemáticos para eliminar las restricciones de las ecuaciones dinámicas, obteniéndose un sistema más simple de ecuaciones. Las restricciones impuestas por las articulaciones y otros componentes mecánicos son uno de los rasgos característicos de los robots, así que no es sorprendente que el formalismo del Lagrange es a menudo el método de opción en la literatura de la robótica.

El método de Euler - Lagrange es muy explícito y sistemático, pudiendo estructurarse en forma algorítmica para los fines de computación, pero tiene el inconveniente de su alta ineficiencia computacional, cuando el orden de las operaciones aritméticas es muy grande. Sin embargo, debido a que las articulaciones de un robot móvil rodante conforman un número pequeño, no existe un excesivo cálculo computacional. Aquí se necesita determinar los pares que se han de aplicar a las ruedas del robot de tal forma que pueda vencer, la inercia propia de cada rueda, la inercia de acoplamiento debido a las interacciones entre las ruedas y el efecto de la gravedad que posteriormente se elimina. Estos pares representan las ecuaciones de movimiento del mecanismo y serán obtenidas a partir del formalismo de la mecánica de Euler - Lagrange. La aplicación del método de las ecuaciones de Euler - Lagrange para elaborar un modelo dinámico consiste en hallar el Lagrangeano total del sistema, basado en la determinación de la energía cinética y potencial de todas las masas del sistema concentradas en forma puntual en el chasis del robot, las ruedas y sus agregados.

4.3 Modelos cinemáticos para las plataformas de aplicación diseñadas y construidas.

4.3.1 Robots tipo Uniciclo.

Para este caso, se asume que el robot móvil se desplazará en una superficie horizontal plana, donde $R = (X, Y, Z)$ es un plano fijo con el eje Z vertical. La posición de la plataforma es dada por un vector q_p de $m_p = 3$ coordenadas que definen la posición y orientación de la plataforma móvil. Esto es escrito como: $q_p = [x \ y \ \Psi]^T$, donde x e y representan la posición en R y Ψ es la orientación de la plataforma móvil. El conjunto de todas las posiciones constituye el *espacio operacional de la plataforma* denominado M_p .

Las plataformas móviles con ruedas pueden ser clasificadas en cuatro categorías:

- 1) Las ruedas fijas, que tienen el eje de la dirección fijo.
- 2) Las ruedas de dirección, en las que el eje pasa a estar en el centro de las dos ruedas.
- 3) Las ruedas castor, donde el eje de orientación no pasa por el centro de la rueda
- 4) Las ruedas suecas, que son similares a las ruedas fijas con la excepción de que usan un parámetro adicional para definir la dirección con respecto al plano de la rueda, al componente cero de la velocidad en el punto de contacto.

Una plataforma robótica Uniciclo, es un robot motorizado que puede girar libremente alrededor de los ejes; el término Uniciclo es usado en robótica para generalizar un vehículo que se mueve libremente en un sistema de referencia bidimensional, conocido también como: tipo uniciclo o comparado uniciclo. Teóricamente son mostrados como vehículos que tienen dos ruedas de tracción paralelas a cada lado de su estructura y presumiblemente una rueda de tipo castor (o dos) que se emplean para su estabilización; esta estructura les permite realizar movimientos simultáneos de rotación y traslación, usando únicamente la rotación de una o ambas ruedas de forma simultánea; una configuración alternativa es usar una sola rueda de tracción que determine la dirección del movimiento de la plataforma y un par de ruedas locas que permitan balancear la estructura y permitan la aplicación correcta del torque en el movimiento.

Si la posición de la plataforma móvil tipo uniciclo está definida por un punto situado delante del centro del eje de las ruedas como se muestra en la Figura 4.2., entonces la configuración cinemática instantánea de la plataforma móvil holonómica se define como:

$$\begin{cases} \dot{x} = u \cos \Psi - a \omega \sin \Psi \\ \dot{y} = u \sin \Psi + a \omega \cos \Psi \\ \dot{\Psi} = \omega \end{cases} \quad (4.1)$$

donde el control (de maniobrabilidad) de la plataforma móvil está definido por $V_p = [u \ \omega]^T$ en la cual u y ω representan las velocidades lineal y angular del robot móvil respectivamente.

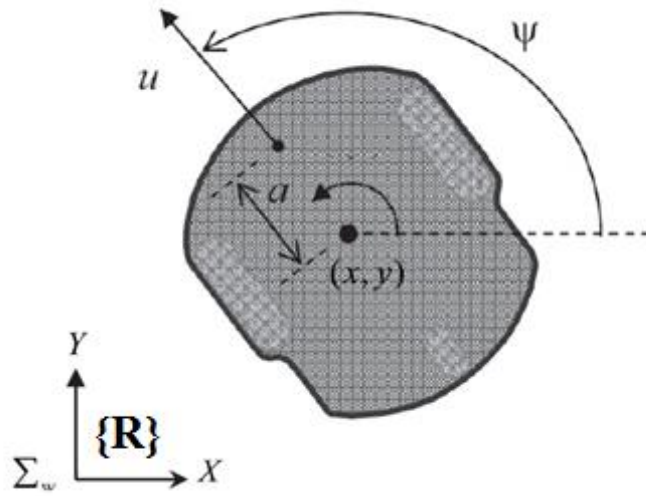


Figura 4.2. Plataforma holonómica Uniciclo

Al considerar una plataforma uniciclo que tenga una distribución de masa no uniforme, como se muestra en la Figura 4.3, se deberá modificar el modelo cinemático de la siguiente forma:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos \Psi & -d \cos \gamma \sin \Psi \\ \sin \Psi & d \cos \gamma \sin \Psi \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ \omega \end{bmatrix} \quad (4.2)$$

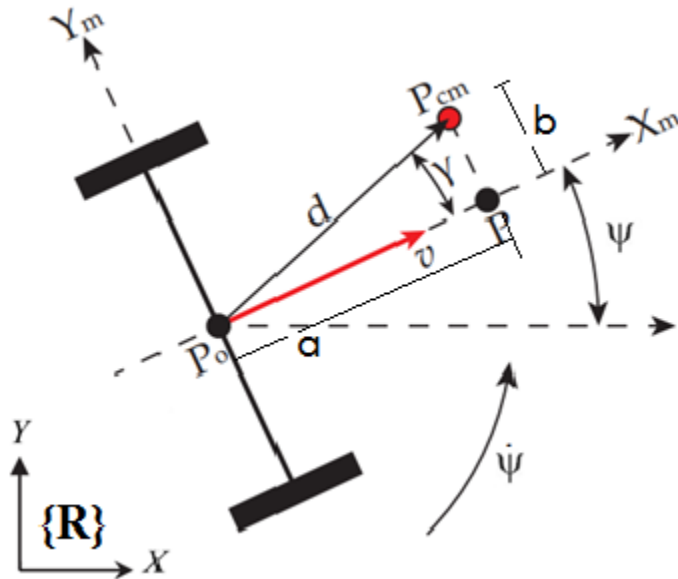


Figura 4.3. Velocidades en robot móvil con distribución de masa no uniforme

$$\begin{aligned}
\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} &= P_0 + R(z) \begin{bmatrix} a \\ b \end{bmatrix} \\
R(z) &= \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \\
\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} &= \begin{bmatrix} \mu \cos \psi - \omega(a \sin \psi + b \cos \psi) \\ \mu \sin \psi + \omega(a \cos \psi - b \sin \psi) \\ \omega \end{bmatrix}
\end{aligned} \tag{4.3}$$

Otro aspecto cinemático que se ha de considerar en este tipo de robot, son las restricciones de tipo no holonómicas que se presentan, en el caso de la Figura 4.3.

4.3.2 Robots Omnidireccionales.

Son tipos de vehículos que pueden moverse en cualquier dirección y desde un punto de origen arbitrario, sin realizar rotaciones previas a sus movimientos, todo sobre un plano de coordenadas cartesianas, proporcionando al sistema tres grados de libertad, movimientos en X, en Y, y la inclinación u orientación ψ . Su principal componente es la forma de accionamiento de los actuadores, ya que los movimientos de deslizamiento por así decirlo, responden a leyes diferentes a las del movimiento de las ruedas del robot unicycle, que proporciona el movimiento al robot (el movimiento de la rueda no es necesariamente el movimiento del motor).

Un robot omnidireccional está conformado por dos velocidades lineales μ_l , μ_m y una velocidad angular ω representadas en el espacio $R(X,Y)$ tal como se indica en la Figura 4.4.

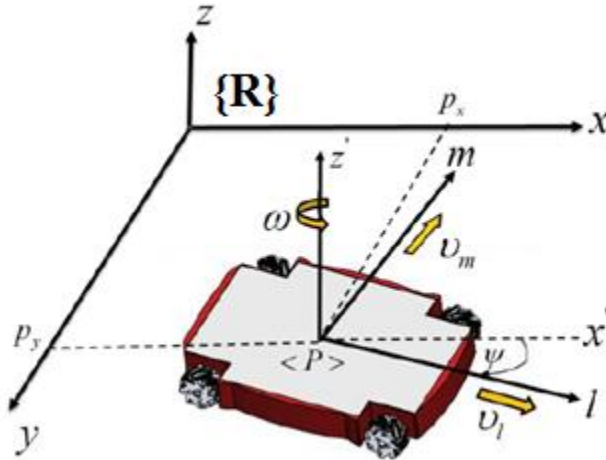


Figura 4.4. Esquemático de la plataforma móvil omnidireccional en el campo inercial $\{R\}$

En este tipo de plataforma omnidireccional, el modelo cinemático está definido de la siguiente forma, tomando en consideración que los puntos de interés se establecen en el centro del robot, tal como se observa en la Figura 4.4; y se establece de la siguiente forma:

$$\begin{bmatrix} \dot{h}_x \\ \dot{h}_y \\ \dot{h}_\psi \end{bmatrix} = \begin{bmatrix} \cos(\Psi) & -\sin(\Psi) & 0 \\ \sin(\Psi) & \cos(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu_l \\ \mu_m \\ \omega \end{bmatrix} \quad (4.4)$$

donde: h_x , h_y y h_ψ son las medidas todas referidas al marco inercial $\{R\}$, de tal forma que el sistema de ecuaciones (4.4) puede escribirse también como:

$$\dot{\mathbf{h}}(t) = \mathbf{J}(\Psi)\mu(t) \quad (4.5)$$

en el cual: $\mathbf{J}(\Psi) \in R^{m \times n}$ donde $m = n = 3$ y representa la matriz Jacobiana que define un mapeo lineal entre el vector de velocidad del robot omnidireccional $\mu \in R^n$ en que $\mu = [\mu_l \ \mu_m \ \omega]^T$ y el vector de velocidad del punto de interés $\dot{\mathbf{h}} \in R^m$ y $\dot{\mathbf{h}} = [\dot{h}_x \ \dot{h}_y \ \dot{h}_\psi]^T$.

Para determinar las velocidades angulares que corresponden a cada rueda y las velocidades lineal y angular del robot omnidireccional se usa el siguiente sistema para las dos transformaciones; a partir de la Figura 4.5.

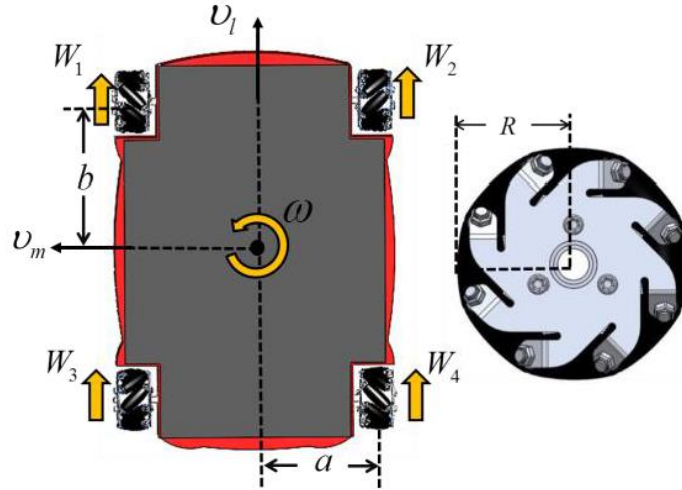


Figura 4.5. Configuración mecanum del robot omnidireccional.

Para encontrar las velocidades del punto de interés generado por el robot omnidireccional se tiene en cuenta cómo el robot será desplazado con cada dirección de rotación de las ruedas omnidireccionales, obteniendo así el sistema de ecuaciones:

$$\mu_l = \left(\frac{W_1 + W_2 + W_3 + W_4}{4} \right) R \quad (4.6)$$

$$\mu_m = \left(\frac{-W_1 + W_2 - W_3 + W_4}{4} \right) \frac{R}{a+b} \quad (4.7)$$

$$\begin{bmatrix} \mu_l \\ \mu_m \\ \omega \end{bmatrix} = \frac{R}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ -\frac{1}{a+b} & \frac{1}{a+b} & -\frac{1}{a+b} & \frac{1}{a+b} \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ W_4 \end{bmatrix} \quad (4.8)$$

es así que (4.8) puede ser escrita de forma matricial como:

$$\boldsymbol{\mu}(t) = \mathbf{J}_T(a, b) \mathbf{W}(t) \quad (4.9)$$

donde: $\mathbf{W} \in R^{4 \times 1}$ es el vector de las velocidades angulares de las ruedas del robot omnidireccional, $\mathbf{J}_T \in R^{4 \times 3}$ es una matriz de transformación, que es obtenida de los parámetros cinemáticos de la plataforma omnidireccional y $\boldsymbol{\mu} \in R^{3 \times 1}$ representa a las velocidades de la plataforma robótica completa.

Las velocidades angulares W_i corresponden a cada rueda de la plataforma y son calculadas en base a las velocidades del movimiento del robot y pueden ser generadas por un controlador o ingresadas como parámetros de entrada por un usuario de forma interactiva; v_{ref} se obtiene de (4.8):

$$\mathbf{W}_d(t) = \mathbf{J}_T^{-1} \boldsymbol{\mu}_{ref}(t) \quad (4.10)$$

Por lo tanto, el sistema de ecuaciones se define de la siguiente manera:

$$\begin{bmatrix} W_{d1} \\ W_{d2} \\ W_{d3} \\ W_{d4} \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & -1 & -(a+b) \\ 1 & 1 & (a+b) \\ 1 & 1 & -(a+b) \\ 1 & -1 & (a+b) \end{bmatrix} \begin{bmatrix} \mu_{lref} \\ \mu_{mref} \\ \omega_{ref} \end{bmatrix} \quad (4.11)$$

donde: $\mathbf{J}_T^{-1} \in R^{4 \times 3}$ representa la matriz de transformación inversa, la cual, fue obtenida de la transformación inversa del robot omnidireccional.

4.3.3 Robots Manipuladores.

La Figura 4.6 muestra la configuración del brazo robótico que se utilizará en las aplicaciones demostrativas, donde $\mathbf{h}_a(t)$ representa la posición del punto de interés (extremo operativo); y $\{R_a\}$ representa el sistema de referencia de coordenadas locales para el brazo robótico; siendo (h_{xa}, h_{ya}, h_{za}) las coordenadas del punto h_a respecto a $\{R_a\}$. Mediante la interpretación geométrica de la cadena de eslabones se determina la posición del extremo operativo en el sistema de referencia $\{R_a\}$.

$$\mathbf{h}_a(t) = \begin{bmatrix} h_{ax} \\ h_{ay} \\ h_{az} \end{bmatrix} = \begin{bmatrix} l_2 \cos(q_2) \cos(q_1) + l_3 \cos(q_2 + q_3) \cos(q_1) \\ l_2 \cos(q_2) \sin(q_1) + l_3 \cos(q_2 + q_3) \sin(q_1) \\ l_1 + l_2 \sin(q_2) + l_3 \sin(q_2 + q_3) \end{bmatrix} \quad (4.12)$$

donde l_1 , l_2 y l_3 son dimensiones del brazo robótico; q_1 , q_2 y q_3 corresponden a los ángulos de giro para cada grado de libertad del robot manipulador.

derivando (4.12) se obtiene el modelo cinemático del robot manipulador, en donde $\dot{\mathbf{h}}_a(t)$ presenta las velocidades de cada articulación en coordenadas de $\{R_a\}$.

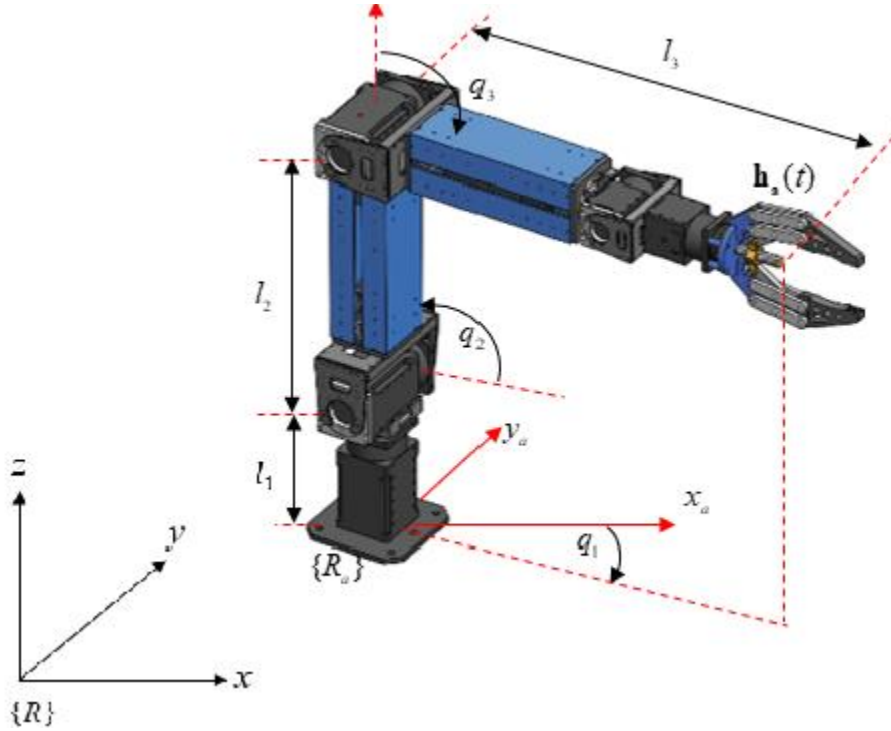


Figura 4.6. Brazo robótico en tres grados de libertad.

$$\begin{bmatrix} \dot{h}_{ax} \\ \dot{h}_{ay} \\ \dot{h}_{az} \end{bmatrix} = \begin{cases} -(l_2 C_2 S_1 + l_3 C_{23} S_1) \dot{q}_1 - (l_2 C_1 S_2 + l_3 S_{23} C_1) \dot{q}_2 - (l_3 S_{23} C_1) \dot{q}_3 \\ (l_3 C_{23} C_1 + l_2 C_1 C_2) \dot{q}_1 - (l_3 S_{23} S_1 + l_2 S_1 S_2) \dot{q}_2 - (l_3 S_{23} S_1) \dot{q}_3 \\ (l_3 C_{23} + l_2 C_2) \dot{q}_2 + (l_3 C_{23}) \dot{q}_3 \end{cases} \quad (4.13)$$

de forma matricial (4.13) puede expresarse:

$$\begin{bmatrix} \dot{h}_{ax} \\ \dot{h}_{ay} \\ \dot{h}_{az} \end{bmatrix} = \begin{bmatrix} -l_2 C_2 S_1 - l_3 C_{23} S_1 & -l_2 C_1 S_2 - l_3 S_{23} C_1 & -l_3 S_{23} C_1 \\ l_1 C_1 C_2 + l_3 C_{23} C_1 & l_3 S_{23} S_1 - l_2 S_1 S_2 & l_3 S_{23} S_1 \\ 0 & l_3 C_{23} + l_2 C_2 & l_3 C_{23} \end{bmatrix} \mathbf{h}_a(t) = \mathbf{J}_a(q_a) \dot{q}_a(t) \quad (4.14)$$

En la que: $\dot{q}_a(t)$ es el vector de velocidades angulares en las articulaciones del manipulador y $\mathbf{J}_a(q_a)$ es la matriz Jacobiana, en función de los ángulos de rotación.

4.3.4 Robots Manipuladores Móviles

Para obtener la modelación cinemática de todo el robot manipulador móvil, se deben emplear los parámetros de la plataforma móvil que se consideraron en forma independiente y del brazo robótico, esto se indica en la Figura 4.7 y el resultado del acoplamiento de ambas plataformas en la Figura 4.8; Es en base a esto que se realiza la modelación cinemática del manipulador móvil:

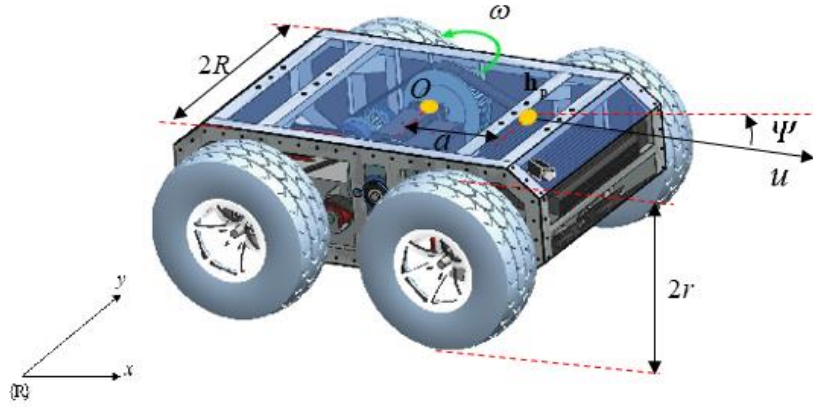


Figura 4.7. Nomenclatura de la plataforma móvil para construir el manipulador móvil

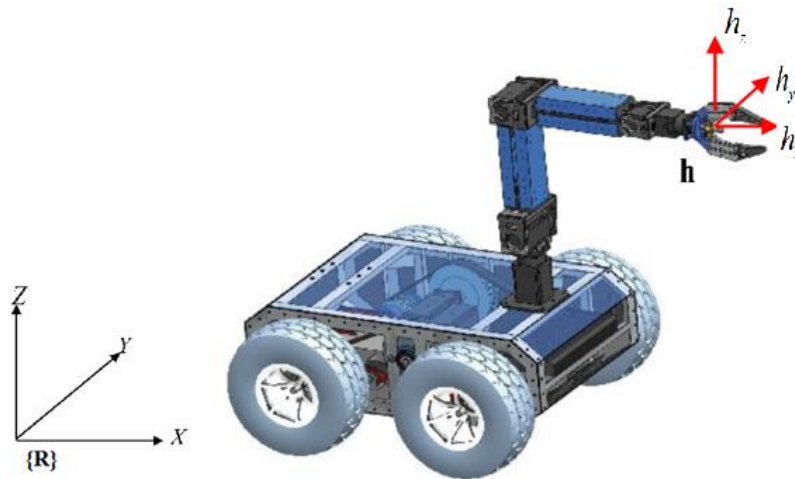


Figura 4.8. Manipulador móvil

En la Figura 4.8 se observa la configuración del manipulador móvil en donde $\mathbf{h}(\mathbf{t})$ representa la posición del extremo operativo del robot, para efectos de las aplicaciones será conocido siempre como efector; todo referido al sistema de referencia inercial $\{R\}$.

El modelo geométrico del manipulador móvil para su extremo operativo mencionado, se obtienen de la siguiente forma:

$$\begin{cases} h_x \\ h_y \\ h_z \end{cases} = \begin{cases} x_o + (a + h_{ax}) \cos(\theta) - h_{ay} \sin(\theta) \\ y_o + (a + h_{ax}) \sin(\theta) + h_{ay} \cos(\theta) \\ h_{az} \end{cases} \quad (4.15)$$

Derivando esta forma matemática, se obtiene el modelo cinemático para el manipulador móvil:

$$\dot{\mathbf{h}}(\mathbf{t}) = \begin{cases} \dot{h}_x \\ \dot{h}_y \\ \dot{h}_z \end{cases} = \begin{cases} \dot{x}_o - (a + h_{ax}) \dot{\theta} \sin(\theta) - h_{ay} \dot{\theta} \cos(\theta) + \dot{h}_{ax} \cos(\theta) - \dot{h}_{ay} \sin(\theta) \\ \dot{y}_o + (a + h_{ax}) \dot{\theta} \cos(\theta) - h_{ay} \dot{\theta} \sin(\theta) + \dot{h}_{ax} \sin(\theta) + \dot{h}_{ay} \cos(\theta) \\ \dot{h}_{az} \end{cases} \quad (4.16)$$

Considerando la posterior modelación dinámica a ser obtenida para este robot, se añade un ángulo de rotación de la plataforma móvil a este modelo, definiendo como $\dot{\mathbf{h}}(t)$ el vector de velocidades lineales desde su extremo en el campo $\{R\}$ y la velocidad de giro \dot{h}_θ del robot móvil independiente. Esto se observa como:

$$\dot{\mathbf{h}}(t) = \begin{cases} \dot{h}_x \\ \dot{h}_y \\ \dot{h}_\theta \\ \dot{h}_z \end{cases} = \begin{cases} \dot{x}_o - (a + h_{ax}) \dot{\theta} \sin(\theta) - h_{ay} \dot{\theta} \cos(\theta) + \dot{h}_{ax} \cos(\theta) - \dot{h}_{ay} \sin(\theta) \\ \dot{y}_o + (a + h_{ax}) \dot{\theta} \cos(\theta) - h_{ay} \dot{\theta} \sin(\theta) + \dot{h}_{ax} \sin(\theta) + \dot{h}_{ay} \cos(\theta) \\ \dot{\theta} \\ \dot{h}_{az} \end{cases} \quad (4.16)$$

De esta forma se obtiene el modelo cinemático compacto para el robot manipulador móvil que se ha construido a partir de la plataforma robótica móvil descrita, este se observa en la Figura 4.7

$$\dot{\mathbf{h}}(\mathbf{t}) = \begin{bmatrix} C_\theta & -(a + h_{ax})S_\theta - h_{ay}C_\theta & (-l_2C_2S_1 - l_3C_{23}S_1)C_\theta - .. & (-l_2C_1S_2 - l_3S_{23}C_1)C_\theta + .. & -l_3S_{23}C_1C_\theta + .. \\ & & ..(l_2C_1C_2 + l_3C_{23}C_1)S_\theta & ..(l_3S_{23}S_1 + l_2S_1S_{21})S_\theta & ..l_3S_{23}S_1S_\theta \\ S_\theta & (a + h_{ax})C_\theta - S_\theta & (-l_2C_2S_1 - l_3C_{23}S_1)S_\theta + .. & (-l_2C_1S_2 - l_3S_{23}C_1)S_\theta + .. & -l_3S_{23}C_1C_\theta - .. \\ & & ..(l_2C_1C_2 + l_3C_{23}C_1)C_\theta & ..(-l_3S_{23}S_1 - l_2S_1S_{21})C_\theta & ..l_3S_{23}S_1S_\theta \\ 0 & 0 & 0 & l_3C_{23} + l_2C_2 & l_3C_{23} \end{bmatrix} \begin{bmatrix} u \\ \omega \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix}$$

$$\dot{\mathbf{h}}(t) = \mathbf{J}(q)\mathbf{v}(t)$$

En la cual $v(t)$ corresponde a todas las velocidades lineales y angulares de la plataforma móvil $[u \ \omega]^T$ y las velocidades angulares de las articulaciones del brazo $[\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3]^T$ y $J(q)$ es la matriz Jacobiana en función de los ángulos de rotación $[\psi \ q_1 \ q_2 \ q_3]^T$.

4.4 Modelos dinámicos para las plataformas de aplicación diseñadas y construidas.

El modelo matemático que representa la dinámica de un robot móvil, un brazo robótico o un manipulador móvil puede obtenerse a partir de las ecuaciones dinámicas de Lagrange, que se basan en la diferencia entre la energía cinética y la energía potencial de cada una de las articulaciones del robot o elementos relacionados en el robot, según el tipo que corresponda.

4.4.1 Robots Uniciclo.

En la obtención del modelo dinámico de un robot móvil uniciclo con restricciones no holonómicas se obtiene a partir de la ecuación de Rosenberg:

$$\frac{\partial}{\partial t} \left(\frac{\partial K}{\partial \dot{q}_i} \right) - \frac{\partial K}{\partial q_i} = Q_i - a_{1i} \lambda_1, \quad i = 1, \dots, 3 \quad (4.17)$$

donde K es la energía cinética, Q_i son los pares de entrada del sistema, a_{1i} son los elementos de la matriz de restricciones no holonómicas y λ_1 es un multiplicador de Lagrange. Haciendo referencia a la Figura 4.3; y utilizando la ecuación de Lewis que usa las velocidades indicadas, la energía cinética del robot móvil se obtiene:

$$K = \frac{1}{2} m v_{cm}^T v_{cm} + \frac{1}{2} \dot{\theta}^2 I \quad (4.18)$$

donde:

$$v_{cm} = \begin{bmatrix} \dot{x}_o - d \sin(\theta + \gamma) \dot{\theta} \\ \dot{y}_o + d \cos(\theta + \gamma) \dot{\theta} \\ 0 \end{bmatrix} \quad (4.19)$$

Considerando que v_{cm} es el centro de masa del robot que estará representada en términos de $q = [x_o \ y_o \ \theta]^T$ de tal forma que el modelo dinámico se obtiene reemplazando (4.18) en (4.17):

$$M(q)\ddot{q} + V_m(q, \dot{q})\dot{q} = B(q)\tau - A^T(q)\lambda \quad (4.20)$$

en el cual:

$$M = \begin{bmatrix} m & 0 & -md \sin(\theta + \gamma) \\ 0 & m & md \cos(\theta + \gamma) \\ -md \sin(\theta + \gamma) & md \cos(\theta + \gamma) & md^2 + I \end{bmatrix} \quad (4.21)$$

$$V_m = \begin{bmatrix} 0 & 0 & -md \cos(\theta + \gamma)\dot{\theta} \\ 0 & 0 & -md \sin(\theta + \gamma)\dot{\theta} \\ 0 & 0 & 0 \end{bmatrix} \quad (4.22)$$

$$B = \frac{1}{\gamma} \begin{bmatrix} \cos(\theta) & \cos(\theta) \\ \sin(\theta) & \sin(\theta) \\ R - d \sin(\gamma) & -R - d \sin(\gamma) \end{bmatrix} \quad (4.23)$$

$$A = [-\sin(\theta) \cos(\theta) \quad -d \cos(\gamma)] \quad (4.24)$$

$$\tau = \begin{bmatrix} \tau_i \\ \tau_d \end{bmatrix} \quad (4.25)$$

donde τ_i y τ_d son los pares de entrada de las ruedas, m es la masa del robot con todas sus características e I es el momento de inercia en P_{cm} alrededor del eje Z_m .

El sistema presentado debe ser no holonómico, para convertirlo en un sistema totalmente controlado, es necesario representar las restricciones no holonómicas como:

$$A(q)\dot{q} = 0 \quad (4.26)$$

Además, que se puede encontrar una matriz $S(q)$ que permita que sus columnas se encuentren en el espacio nulo de $A(q)$ de tal modo que:

$$S^T(q)A^T(q) = 0 \quad (4.27)$$

Así también que la velocidad \dot{q} se encuentra también en el espacio nulo de $A(q)$ y existe un vector η que cumpla:

$$\dot{q} = S(q)\eta \quad (4.28)$$

Que está definido por: $\eta = [v \quad \omega]^T$ que obliga a que la (4.28) se relacione con el modelo cinemático del robot en el punto P_{cm} y la matriz $S(q)$ este dada por:

$$S(q) = \begin{bmatrix} \cos(\theta) & -d \sin(\theta + \gamma) \\ \sin(\theta) & d \cos(\theta + \gamma) \\ 0 & 1 \end{bmatrix} \quad (4.29)$$

Sustituyendo (4.28) en (4.20) se obtiene el modelo dinámico más apropiado, multiplicando por la izquierda por $S(q)^T$:

$$S^T M(S\dot{\eta} + \dot{S}\eta) + S^T V_m S\eta = S^T B\tau \quad (4.30)$$

4.4.2 Robot Omnidireccional.

El modelo dinámico para el robot móvil omnidireccional de configuración mecanum construido puede ser calculado con la siguiente ecuación de forma general y universal:

$$R_\omega^R(\theta)[T]\xi + E^T[T]\psi = E^T\tau_\varphi \quad (4.31)$$

donde:

$$[T]_\psi = \frac{\partial}{\partial t} \left(\frac{\partial T}{\partial \dot{\psi}} \right) - \frac{\partial T}{\partial \psi} \quad (4.32)$$

y:

$$\tau_\varphi = [\tau_{\varphi 1} \quad \tau_{\varphi 2} \quad \tau_{\varphi 3} \quad \tau_{\varphi 4}]^T \quad (4.33)$$

Los términos que corresponden son: T es la suma de todas las energías de tipo cinéticas presentes en el robot y $\tau_{\varphi i}$ corresponde al torque aplicado en cada una de las ruedas; Si m_R es la masa total del robot, sumando las masas de su chasis y sus ruedas, I_{Rz} es la inercia considerada total del robot, incluidos también las ruedas y el chasis; y finalmente $I_{\varphi y}$ la inercia de las ruedas en el eje de giro, esta energía se puede expresar de la siguiente forma también:

$$T = \dot{\xi}^T (R_\omega^R(\theta))^T M R_\omega^R(\theta) \dot{\xi} + \dot{\varphi}^T I_\varphi \dot{\varphi} \quad (4.34)$$

que cumple:

$$M = \frac{1}{2} \text{diag}\{m_R \quad m_R \quad I_{Rz}\} \quad (4.35)$$

$$I_\varphi = \frac{1}{2} \text{diag}\{I_{\varphi y} \quad I_{\varphi y} \quad I_{\varphi y} \quad I_{\varphi y}\} \quad (4.36)$$

$$\dot{\varphi} = [\dot{\varphi}_1 \quad \dot{\varphi}_2 \quad \dot{\varphi}_3 \quad \dot{\varphi}_4] \quad (4.37)$$

al reemplazar en (4.34) se encuentra que:

$$T = \frac{m_R \dot{x}^2}{2} + \frac{m_R \dot{y}^2}{2} + \frac{I_{Rz} \dot{\theta}^2}{2} + \frac{I_{\varphi y}}{2} (\dot{\varphi}_1^2 + \dot{\varphi}_2^2 + \dot{\varphi}_3^2 + \dot{\varphi}_4^2) \quad (4.38)$$

obteniéndose también:

$$[T]_{\xi} = \frac{\partial}{\partial t} \left(\frac{\partial T}{\partial \dot{\xi}} \right) - \frac{\partial T}{\partial \xi} \quad (4.39)$$

resultando:

$$[T]_{\xi} = M_R \ddot{\xi} \quad (4.40)$$

con:

$$M_R = \text{diag}\{m_R, m_R, I_{Rz}\} \quad (4.41)$$

$$\ddot{\xi} = [\ddot{x} \quad \ddot{y} \quad \ddot{\theta}]^T \quad (4.42)$$

de igual forma usando (4.34) se tiene:

$$[T]_{\varphi} = \frac{\partial}{\partial t} \left(\frac{\partial T}{\partial \dot{\varphi}} \right) - \frac{\partial T}{\partial \varphi} \quad (4.43)$$

igualmente:

$$[T]_{\varphi} = M_{\varphi} \ddot{\varphi} \quad (4.44)$$

siendo:

$$M_{\varphi} = \text{diag}\{I_{\varphi y}, I_{\varphi y}, I_{\varphi y}, I_{\varphi y}\} \quad (4.45)$$

$$\ddot{\varphi} = [\ddot{\varphi}_1, \ddot{\varphi}_2, \ddot{\varphi}_3, \ddot{\varphi}_4] \quad (4.46)$$

sustituyendo (4.43) y (4.39) en (4.31):

$$R_{\omega}^R(\theta) M_R \ddot{\xi} + E^T M_{\varphi} \ddot{\varphi} = E^T \tau_{\varphi} \quad (4.47)$$

convirtiendo las aceleraciones al plano del robot:

$$\ddot{\xi} = R_R^{\omega}(\theta) \dot{\eta} + \dot{R}_R^{\omega}(\theta) \eta \quad (4.48)$$

Las aceleraciones de las ruedas se obtienen derivando esta:

$$\ddot{\varphi} = E \dot{\eta} \quad (4.49)$$

Sustituyendo (4.49) y (4.48) en (4.47):

$$\tilde{M} \dot{\eta} + \tilde{C} \eta = E^T \tau_{\varphi} \quad (4.50)$$

Que refieren a la matriz de inercia y a la matriz de fuerzas centrípetas y centrífugas en las ruedas del conjunto móvil; y se definen:

$$\tilde{M} = R_{\omega}^R(\theta)M_R R_{\omega}^{\omega}(\theta) + E^T M_{\phi} E \quad (4.51)$$

$$\tilde{C} = R_{\omega}^R(\theta)M_R \dot{R}_{\omega}^R(\theta) \quad (4.52)$$

4.4.3 Robot Manipulador.

En la obtención del modelo dinámico del manipulador se hace uso del modelo de Euler – Lagrange, que presenta la facilidad de definir las ecuaciones en función del torque producido en cada una de las articulaciones del brazo robótico, aplicando un balance energético (cinético – potencial); aquí se tomarán en cuenta las masas puntuales en los extremos de cada eslabón, debido al gran peso que presentan los actuadores.

Con el fin de determinar la energía cinética es necesario encontrar las velocidades a las cuales se mueven los eslabones con respecto al sistema de referencia $\{R_a\}$, iniciando en las ecuaciones de posición para el extremo de cada eslabón ($\mathbf{h}_{a2}, \mathbf{h}_{a3}$).

$$\mathbf{h}_{a2} = \begin{bmatrix} h_{ax2} \\ h_{ay2} \\ h_{az2} \end{bmatrix} = \begin{bmatrix} l_2 \cos(q_2) \cos(q_1) \\ l_2 \cos(q_2) \sin(q_1) \\ h + l_2 \sin(q_2) \end{bmatrix} \quad (4.53)$$

$$\mathbf{h}_{a3} = \begin{bmatrix} h_{ax3} \\ h_{ay3} \\ h_{az3} \end{bmatrix} = \begin{bmatrix} l_2 \cos(q_2) \cos(q_1) + l_3 \cos(q_2 + q_3) \cos(q_1) \\ l_2 \cos(q_2) \sin(q_1) + l_3 \cos(q_2 + q_3) \sin(q_1) \\ h + l_2 \sin(q_2) + l_3 \sin(q_2 + q_3) \end{bmatrix} \quad (4.54)$$

Para encontrar la velocidad para los eslabones en cada extremo se deriva estas ecuaciones obteniendo:

$$\dot{\mathbf{h}}_{a2} = \begin{bmatrix} \dot{h}_{ax2} \\ \dot{h}_{ay2} \\ \dot{h}_{az2} \end{bmatrix} = \begin{bmatrix} -l_2 \cos(q_2) \sin(q_1) \dot{q}_1 - l_2 \sin(q_2) \cos(q_1) \dot{q}_2 \\ l_2 \cos(q_2) \cos(q_1) \dot{q}_1 - l_2 \sin(q_2) \sin(q_1) \dot{q}_2 \\ l_2 \cos(q_2) \dot{q}_2 \end{bmatrix} \quad (4.55)$$

$$\dot{\mathbf{h}}_{a3} = \begin{bmatrix} \dot{h}_{ax3} \\ \dot{h}_{ay3} \\ \dot{h}_{az3} \end{bmatrix} = \begin{bmatrix} \dot{h}_{ax2} - l_3 \cos(q_2 + q_3) \sin(q_1) \dot{q}_1 - l_3 \sin(q_2 + q_3) \cos(q_1) (\dot{q}_2 + \dot{q}_3) \\ \dot{h}_{ay2} - l_3 \cos(q_2 + q_3) \cos(q_1) \dot{q}_1 - l_3 \sin(q_2 + q_3) \sin(q_1) (\dot{q}_2 + \dot{q}_3) \\ \dot{h}_{az2} + l_3 \cos(q_2 + q_3) (\dot{q}_2 + \dot{q}_3) \end{bmatrix} \quad (4.56)$$

Usando las ecuaciones (4.55) y (4.56) se resuelve el valor de la energía cinética del manipulador, encontrada a partir de : $K_a = \frac{1}{2} \dot{h}_{a2}^T m_2 \dot{h}_{a2} + \frac{1}{2} \dot{h}_{a3}^T m_3 \dot{h}_{a3}$ donde \dot{h}_{a2} representa a las velocidades en el primer eslabón del brazo, \dot{h}_{a3} representa las velocidades del eslabón en el extremo operativo; m_2 y m_3 las masas puntuales de los eslabones 2 y 3, se obtiene:

$$K_a = \frac{1}{2} m_2 l_2^2 (\dot{q}_2^2 + \dot{q}_1^2 \cos^2(q_2)) + \frac{1}{2} m_3 \left(\left(l_2 l_3 \cos(q_3) + \frac{l_2^2}{2} + \frac{l_3^2}{2} \right) \dot{q}_2^2 + \frac{l_3^2}{2} \dot{q}_3^2 \dots \right. \\ \left. \dots + \left(\frac{l_2^2 + l_2^2 \cos(2q_2)}{4} \dots \right. \right. \\ \left. \dots + \frac{l_3^2 + l_3^2 \cos(2q_2 + 2q_3)}{4} \dots \right. \\ \left. \dots + \frac{l_2 l_3 \cos(q_3) + l_2 l_3 \cos(2q_2 + q_3)}{2} \right) \dot{q}_1^2 \dots \left. \dots + (l_3^2 + l_2 l_3 \cos(q_3)) \dot{q}_2 \dot{q}_3 \right) \quad (4.57)$$

Con la finalidad de abordar de forma completa el modelo dinámico usando Euler – Lagrange se debe encontrar la energía potencial del conjunto de todos los eslabones, según $P_a = m_2 g h_{az2} + m_3 g h_{az3}$ en la cual h_{az2} y h_{az3} son las posiciones en el eje Z de los eslabones 2 y 3.

$$P_a = (m_2 + m_3) g h + m_2 g l_2 \sin(q_2) + m_3 g l_2 \sin(q_2) + m_3 g l_3 \sin(q_2 + q_3) \quad (4.58)$$

Para encontrar el Lagrangeano que define la aplicación del método de Euler – Lagrange, se resta la energía cinética menos la energía potencial, y se obtiene:

$$\begin{aligned}
L_a = & \left(\frac{2l_2^2 m_2 \cos(q_2)^2 + 2l_2^2 m_3 \cos(q_2)^2 + 2l_2 l_3 m_3 \cos(2q_2 + q_3) \dots}{4} \right) \dot{q}_1^2 \dots \\
& \dots + \left(\frac{l_2^2 m_3 + l_3^2 m_3 + 2l_2 l_3 m_3 \cos(q_3) + l_2^2 m_2}{2} \right) \dot{q}_2^2 \\
& + (l_2 l_3 m_3 \cos(q_3) + l_3^2 m_3) \dot{q}_2 \dot{q}_3 + \frac{l_3^2 m_3 \dot{q}_3^2}{2} \dots \\
& \dots - g h(m_2 + m_3) - g l_3 m_3 \sin(q_2 + q_3) - g l_2 m_2 \sin(q_2) - g l_2 m_3 \sin(q_2)
\end{aligned} \tag{4.59}$$

Para determinar el torque de cada articulación, el método descrito, lo hace usando:

$$\frac{\partial}{\partial t} = \left(\frac{\partial L_a}{\partial \dot{q}_i} \right) - \frac{\partial L_a}{\partial q_i} = \tau_i$$

resolviendo esta derivada:

$$\frac{\partial L}{\partial \dot{q}_1} = \left(l_3^2 m_3 \cos(q_2 + q_3)^2 + l_2^2 m_2 \cos(q_2)^2 + l_2 m_3 \cos(q_2)^2 \dots \right) \dot{q}_1 \tag{4.60}$$

$$\frac{\partial L}{\partial \dot{q}_2} = (l_2^2 m_2 + l_2^2 m_3 + l_3^2 m_3 + 2l_2 l_3 m_3 \cos(q_3)) \dot{q}_2 + (l_3^2 m_3 + l_2 l_3 m_3 \cos(q_3)) \dot{q}_3 \tag{4.61}$$

$$\frac{\partial L}{\partial \dot{q}_3} = l_3 m_3 (l_3 \dot{q}_2 + l_3 \dot{q}_3 + l_2 \dot{q}_2 \cos(q_3)) \tag{4.62}$$

Estas ecuaciones se derivan respecto al tiempo y se obtienen:

$$\begin{aligned}
\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_1} \right) = & \left(\frac{l_3^2 m_3 \cos(q_2 + q_3)^2 + l_2^2 m_2 \cos(q_2)^2 + l_2^2 m_3 \cos(q_2)^2 \dots}{\dots + 2l_2 l_3 m_3 \cos(q_2 + q_3) \cos(q_2)} \right) \ddot{q}_1 \dots \\
& \dots + 2 \left(\frac{-l_2^2 m_2 \cos(q_2) \sin(q_2) - 2l_2^2 m_3 \cos(q_2) \sin(q_2)}{-l_3^2 m_3 \cos(q_2 + q_3) \sin(q_2 + q_3) - l_2 l_3 m_3 \cos(q_2 + q_3) \sin(q_2) \dots} \right) \dot{q}_1 \dot{q}_2 \dots \\
& \dots + (-2l_3^2 m_3 \cos(q_2 + q_3) \sin(q_2 + q_3) - 2l_2 l_3 m_3 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1 \dot{q}_3
\end{aligned} \tag{4.63}$$

$$\begin{aligned}
\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_2} \right) = & (l_2^2 m_2 + l_2^2 m_3 + l_3^2 m_3 + 2l_2 l_3 m_3 \cos(q_3)) \ddot{q}_2 + (l_3^2 m_3 + l_2 l_3 m_3 \cos(q_3)) \ddot{q}_3 \\
& - l_2 l_3 m_3 \dot{q}_3^2 \sin(q_3) - 2l_2 l_3 m_3 \dot{q}_2 \dot{q}_3 \sin(q_3)
\end{aligned} \tag{4.64}$$

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_3} \right) = l_3 m_3 (l_3 \ddot{q}_2 + l_3 \ddot{q}_3 + l_2 \ddot{q}_2 \cos(q_3) + l_2 \dot{q}_2 \dot{q}_3 \sin(q_3)) \quad (4.65)$$

derivando respecto a las posiciones angulares: $\frac{\partial L}{\partial q_1} = 0$

$$\frac{dL}{dq_2} = \left(\frac{-l_2^2 m_2 \dot{q}_1^2 \sin(2q_2) - l_2^2 m_3 \sin(2q_2) \dots}{2} \right) \dot{q}_1^2 \dots \quad (4.66)$$

$$\frac{dL}{dq_3} = l_3 m_3 \left(\begin{aligned} & \dots - gl_3 m_3 \cos(q_2 + q_3) - gl_2 m_2 \cos(q_2) - gl_2 m_3 \cos(q_2) \\ & - g \cos(q_2 + q_3) - \frac{l_3 \sin(2q_2 + 2q_3) + l_2 \sin(q_3) + l_2 \sin(2q_2 + q_3)}{2} \dot{q}_1^2 \dots \\ & \dots - l_2 \dot{q}_2^2 \sin(q_3) - l_2 \dot{q}_2 \dot{q}_3 \sin(q_3) \end{aligned} \right) \quad (4.67)$$

utilizando: $\tau_{a1} = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_1} \right) - \frac{\partial L}{\partial q_1}$ y aplicando a ella $\frac{\partial L}{\partial q_1} = 0$ y (3,63) para encontrar el torque generado en el primer motor del brazo robótico:

$$\begin{aligned} \tau_{a1} = & \left(l_3^2 m_3 \cos(q_2 + q_3)^2 + l_2^2 m_2 \cos(q_2)^2 + l_2^2 m_3 \cos(q_2)^2 \dots \right) \ddot{q}_1 \dots \\ & - 2 \left(l_2^2 m_2 \cos(q_2) \sin(q_2) + l_3^2 m_3 \cos(q_2 + q_3) \sin(q_2 + q_3) + \dots \right. \\ & \left. \dots l_2^2 m_3 \cos(q_2) \sin(q_2) + l_2 l_3 m_3 (\cos(q_2 + q_3) \sin(q_2) + \sin(q_2 + q_3) \cos(q_2)) \right) \dot{q}_1 \dot{q}_2 \\ & + (-2l_3^2 m_3 \cos(q_2 + q_3) \sin(q_2 + q_3) - 2l_2 l_3 m_3 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1 \dot{q}_3 \end{aligned} \quad (4.68)$$

y de la misma manera usando:

$$\begin{aligned} \tau_{a2} &= \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_2} \right) - \frac{\partial L}{\partial q_2} \\ \tau_{a3} &= \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_3} \right) - \frac{\partial L}{\partial q_3} \end{aligned}$$

encontramos el torque generado en los motores 2 y 3:

$$\begin{aligned} \tau_{a2} = & (l_2^2 m_2 + l_2^2 m_3 + l_3^2 m_3 + 2l_2 l_3 m_3 \cos(q_3)) \ddot{q}_2 + (l_3^2 m_3 + l_2 l_3 m_3 \cos(q_3)) \ddot{q}_3 \dots \\ & \dots + \left(\frac{l_2^2 m_2 \sin(2q_2) + l_3^2 m_3 \sin(2q_2 + 2q_3) + l_2^2 m_3 \sin(2q_2) \dots}{2} \right) \dot{q}_1^2 \dots \\ & \dots - l_2 l_3 m_3 \dot{q}_3^2 \sin(q_3) - 2l_2 l_3 m_3 \dot{q}_2 \dot{q}_3 \sin(q_3) + g l_3 m_3 \cos(q_2 + q_3) \dots \\ & \dots + g l_2 m_2 \cos(q_2) + g l_2 m_3 \cos(q_2) \end{aligned} \quad (4.69)$$

$$\tau_{a3} = l_3 m_3 \left(\begin{aligned} & (l_3 + l_2 \cos(q_3)) \ddot{q}_2 + l_3 \ddot{q}_3 + g \cos(q_2 + q_3) + l_2 \dot{q}_2^2 \sin(q_3) \dots \\ & \dots + \frac{l_2 \sin(q_3) + l_2 \sin(2q_2 + q_3) + l_3 \sin(2q_2 + 2q_3)}{2} \dot{q}_1^2 \end{aligned} \right) \quad (4.70)$$

Agrupando (4.68), (4.69) y (4.79) de forma matricial obtenemos el modelo dinámico del brazo robótico: el cual expresado en estos términos es: $\tau_a = [\tau_{a1} \quad \tau_{a2} \quad \tau_{a3}]$ producidos para cada una de las articulaciones, entonces:

$$\tau_a(t) = M_a(q_a) \ddot{q}_a(t) + C_a(q_a, \dot{q}_a) \dot{q}_a(t) + g_a(q_a) \quad (4.71)$$

en la cual $M_a(q)$ es la matriz de inercia; $C_a(q, \dot{q})$ es la matriz de fuerzas centrífugas y centrípetas en los eslabones del manipulador y $g_a(q)$ es el vector gravedad.

4.4.4 Robot Manipulador móvil.

De igual forma como se había explicado en el planteamiento del modelo cinemático del robot manipulador móvil, el modelo dinámico del mismo, se basa en la combinación de los conceptos propuestos para la plataforma independiente y el brazo robótico independiente.

Como consecuencia lógica de esto, se aplica el mismo método de Euler – Lagrange para obtener las ecuaciones de las fuerzas y los torques que intervienen en el sistema, pero además se realiza un conjunto de modificaciones para obtener un modelo dinámico compacto del robot manipulador móvil, en referencia a las variables principales que se considera intervienen.

Por lo tanto, la energía cinética del robot manipulador móvil es definida por:

$$K = \frac{1}{2} \dot{\mathbf{h}}_p^T(t) m \dot{\mathbf{h}}_p(t) + \frac{1}{2} \dot{\mathbf{h}}_2^T m_2 \dot{\mathbf{h}}_2 + \frac{1}{2} \dot{\mathbf{h}}_3^T m_3 \dot{\mathbf{h}}_3 \quad (4.72)$$

donde:

$\dot{\mathbf{h}}_p$ corresponde a las velocidades del robot móvil; $\dot{\mathbf{h}}_2$ corresponde a las velocidades del primer eslabón del brazo y $\dot{\mathbf{h}}_3$ son las velocidades en el eslabón del extremo operativo del brazo.

Además, se define la energía potencial del conjunto completo por: $P = m_2 q h_{z2} + m_3 q h_{z3}$ en la cual h_{z2} y h_{z3} están definidas por las posiciones de los eslabones 2 y 3 en el eje Z.

Las ecuaciones de la energía cinética y potencial del robot manipulador móvil se definen por:

$$\begin{aligned}
 K = & \frac{m}{2} \left((a^2 + 1) \dot{\theta}^2 + 2C_\theta a \dot{\theta} \dot{y} \right) + \left(\frac{m_2}{2} + \frac{m_3}{2} \right) \dot{\theta}^2 + \frac{m_3}{2} (\dot{q}_2 (l_3 C_{23} + l_2 C_2) + l_3 \dot{q}_3 C_{23})^2 \dots \\
 & \dots + \frac{m_3}{2} \left(\dot{q}_1 \left(\frac{S_\theta (l_3 C_{23} C_1 + l_2 C_1 C_2)}{+ C_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1)} \right) - \dot{y}_p \dots \right)^2 \\
 & \dots + \frac{m_3}{2} \left(\dots + \dot{q}_2 \left(\frac{C_\theta (l_3 C_{23} C_1 + l_2 C_1 C_2)}{- S_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1)} \right) \dots \right. \\
 & \dots + \dot{\theta} \left(\frac{C_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1)}{+ S_\theta (a + l_3 C_{23} C_1 + l_2 C_1 C_2)} \right) \dots \\
 & \dots + \dot{q}_3 \left(\frac{l_3 S_{23} C_1 C_\theta - l_3 S_{23} S_1 S_\theta}{+ l_3 S_{23} C_\theta S_1} \right) - \dot{y}_p \left. \right)^2 \\
 & \dots + \frac{m_2}{2} \left(\dot{y}_p - \dot{q}_2 \left(\frac{l_2 C_1 S_2 S_\theta}{+ l_2 S_1 S_2 C_\theta} \right) \right)^2 \\
 & \dots + \frac{m_2}{2} \left(+ \dot{\theta} \left(\frac{a C_\theta + l_2 C_1 C_2 C_\theta}{- l_2 C_2 S_1 S_\theta} \right) \right. \\
 & \dots + \dot{q}_1 \left(\frac{l_2 C_1 C_2 C_\theta}{- l_2 C_2 S_1 S_\theta} \right) \left. \right)^2 \\
 & \dots + \frac{m_2}{2} \left(\dot{q}_2 \left(\frac{l_2 C_1 C_\theta S_2}{- l_2 S_1 S_2 S_\theta} \right) - \dot{y}_p \right)^2 \\
 & \dots + \frac{m_2}{2} \left(+ \dot{\theta} \left(\frac{a S_\theta + l_2 C_1 C_2 S_\theta}{+ l_2 C_2 S_1 C_\theta} \right) \right. \\
 & \dots + \dot{q}_1 \left(\frac{l_2 C_1 C_2 S_\theta}{+ l_2 C_2 S_1 C_\theta} \right) \left. \right)^2 + \frac{l_2^2 m_2}{2} \dot{q}_2^2 C_2^2
 \end{aligned} \tag{4.73}$$

$$P = (m_2 + m_3)gh + m_2 gl_2 \sin(q_2) + m_3 gl_2 \sin(q_2) + m_3 gl_3 \sin(q_2 + q_3) \tag{4.74}$$

Con estas, se aplica el método de Euler – Lagrange restando de la energía cinética la energía potencial, y obteniendo de esta forma el Lagrangeano:

$$\begin{aligned}
L = & \left(\frac{m+m_2+m_3}{2} \right) \dot{x}_p^2 + \left(\frac{m+m_2+m_3}{2} \right) \dot{y}_p^2 + \left(\frac{l_3^2 m_3 C_{23}^2 + l_2^2 m_2 C_2^2 + l_2^2 m_3 C_2^2}{2} \right. \\
& \left. + \frac{2l_2 l_3 m_3 C_{23} C_2}{2} \right) \dot{q}_1^2 + \left(\frac{m+m_2+m_3+a^2 m+a^2 m_2+a^2 m_3+l_3^2 m_3 C_{23}^2+l_2^2 m_2 C_2^2}{2} \right. \\
& \left. + l_2^2 m_3 C_2^2 + al_3 m_3 C_{23} C_1 + l_2 l_3 m_3 C_{23} C_2 + al_2 m_2 C_1 C_2 + al_2 m_3 C_1 C_2 \right) \dot{\theta}^2 + \left(\frac{l_3^2 m_3}{2} \right) \dot{q}_3^2 \\
& + \left(\frac{l_2^2 m_2 + l_2^2 m_3 + l_3^2 m_3 + 2l_2 l_3 m_3 C_3}{2} \right) \dot{q}_2^2 - S_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{x} \\
& - C_{\theta 1} \left(\frac{l_2 m_2 S_2 + l_2 m_3 S_2}{+l_3 m_3 S_{23}} \right) \dot{q}_2 \dot{x} - \left(\frac{m_2 (S_{\theta} (a + l_2 C_1 C_2) + l_2 C_2 C_{\theta} S_1) - am S_{\theta}}{+m_3 \left(\begin{array}{l} C_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ +S_{\theta} (a + l_3 C_{23} C_1 + l_2 \cos(q_1) C_2) \end{array} \right)} \right) \dot{\theta} \dot{x} \\
& - l_3 m_3 C_{\theta 1} S_{23} \dot{q}_3 \dot{x} - l_3 m_3 S_{23} S_{\theta 1} \dot{q}_3 \dot{y} + \left(\frac{m_2 (C_{\theta} (a + l_2 C_1 C_2) - l_2 C_2 S_1 S_{\theta}) + am C_{\theta}}{-m_3 \left(\begin{array}{l} S_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ -C_{\theta} (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \end{array} \right)} \right) \dot{\theta} \dot{y} \\
& + C_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{y} - S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_2 \dot{y} \\
& + \left(\frac{l_3^2 m_3 C_{23}^2 + l_2^2 m_2 C_2^2 + l_2^2 m_3 C_2^2}{+al_3 m_3 C_{23} C_1 + 2l_2 l_3 m_3 C_{23} C_2} \right) \dot{q}_1 \dot{\theta} - a S_1 (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_2 \dot{\theta} \\
& + \left(\frac{+al_2 m_2 C_1 C_2 + al_2 m_3 C_1 C_2}{-al_3 m_3 S_{23} S_1 \dot{q}_3 \dot{\theta} + l_3 m_3 (l_3 + l_2 C_3) \dot{q}_2 \dot{q}_3 + (m_2 (h + l_2 S_2) + m_3 (h + l_3 S_{23} + l_2 S_2)) g} \right)
\end{aligned} \tag{4.75}$$

Esta ecuación define el torque y las fuerzas generadas, en el punto de la plataforma móvil, en el cual está instalado el robot manipulador, determinado por: (x_p, y_p, θ_p) y el torque y velocidades en las articulaciones del brazo (q_1, q_2, q_3) usando: $\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q}$ se resuelve esta derivada con relación a las velocidades lineales y angulares $(\dot{x}_p, \dot{y}_p, \dot{\theta}_p, \dot{q}_1, \dot{q}_2, \dot{q}_3)$ resultando:

$$\begin{aligned}
\frac{dL}{d\dot{x}} = & (m+m_2+m_3) \dot{x} + (l_3 m_3 C_{\theta 1} S_{23}) \dot{q}_3 - S_{\theta 1} \left(\frac{l_2 m_2 C_2 + l_2 m_3 C_2}{+l_3 m_3 C_{23}} \right) \dot{q}_1 \dots \\
& \dots - \left(\frac{m_2 (S_{\theta} (a + l_2 C_1 C_2) + l_2 C_2 C_{\theta} S_1) - am S_{\theta}}{+m_3 \left(\begin{array}{l} C_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ +S_{\theta} (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \end{array} \right)} \right) \dot{\theta} - C_{\theta 1} \left(\frac{l_2 m_2 S_2 + l_2 m_3 S_2}{+l_3 m_3 S_{23}} \right) \dot{q}_2
\end{aligned} \tag{4.76}$$

$$\begin{aligned} \frac{dL}{d\dot{y}_p} = & (m + m_2 + m_3) \dot{y} + (C_{1\theta}(l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23})) \dot{q}_1 - l_3 m_3 S_{23} S_{\theta 1} \dot{q}_3 \dots \\ & \dots + \left(m_2 (C_{\theta}(a + l_2 C_1 C_2) - 2l_2 C_2 S_1 S_{\theta}) + a m C_{\theta} \dots \right) \dot{\theta} - S_{\theta 1} \left(\begin{matrix} l_2 m_2 S_2 + l_2 m_3 S_2 \dots \\ \dots + l_3 m_3 S_{23} \end{matrix} \right) \dot{q}_2 \end{aligned} \quad (4.77)$$

$$\begin{aligned} \frac{dL}{d\dot{\theta}_p} = & \left(\begin{matrix} m + m_2 + m_3 + a^2 m_2 + a^2 m_3 + m a^2 + l_2^2 m_2 C_2^2 \\ + l_2^2 m_3 C_2^2 + l_3^2 m_3 C_{23}^2 + 2a l_3 m_3 C_{23} C_1 + 2a l_2 m_2 C_1 C_2 \\ + 2a l_2 m_3 C_1 C_2 + 2l_2 l_3 m_3 C_{23} C_2 \end{matrix} \right) \dot{\theta} - a l_3 m_3 S_{23} S_1 \dot{q}_3 \dots \\ & \dots + \left(\begin{matrix} l_3^2 m_3 C_{23}^2 + l_2^2 m_2 C_2^2 + l_2^2 m_3 C_2^2 \\ + a l_3 m_3 C_{23} C_1 + 2l_2 l_3 m_3 C_{23} C_2 \\ + a l_2 m_2 C_1 C_2 + a l_2 m_3 C_1 C_2 \end{matrix} \right) \dot{q}_1 + \left(\begin{matrix} m a C_{\theta} + a m_2 C_{\theta} + a m_3 C_{\theta} - l_3 m_3 C_{23} S_1 S_{\theta} \\ + l_3 m_3 C_{23} C_1 C_{\theta} + l_2 m_2 C_1 C_2 C_{\theta} \\ - l_2 m_2 C_2 S_1 S_{\theta} - l_2 m_3 C_2 S_1 S_{\theta} + l_2 m_3 C_1 C_2 C_{\theta} \end{matrix} \right) \dot{y} \dots \\ & \dots + \left(\begin{matrix} -m a S_{\theta} - a m_2 S_{\theta} - a m_3 S_{\theta} - l_3 m_3 C_{23} C_1 S_{\theta} \\ - l_3 m_3 C_{23} C_{\theta} S_1 - l_2 m_3 C_1 \cos(q_2) S_{\theta} \\ - l_2 m_2 C_1 C_2 S_{\theta} - l_2 m_2 C_2 C_{\theta} \sin(q_1) - l_2 m_3 C_2 C_{\theta} S_1 \end{matrix} \right) \dot{x} - \left(\begin{matrix} a l_3 m_3 S_{23} S_1 + a l_2 m_2 S_1 S_2 \\ + a l_2 m_3 S_1 S_2 \end{matrix} \right) \dot{q}_2 \end{aligned} \quad (4.78)$$

$$\begin{aligned} \frac{dL}{d\dot{q}_1} = & -S_{\theta 1}(l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{x} + C_{\theta 1}(l_2 m_2 C_2 + l_2 m_3 \cos(q_2) + l_3 m_3 C_{23}) \dot{y} \dots \\ & \dots + \left(\begin{matrix} l_3^2 m_3 C_{23}^2 + l_2^2 m_2 C_2^2 + l_2^2 m_3 C_2^2 + a l_3 m_3 C_{23} C_1 \\ + 2l_2 l_3 m_3 C_{23} C_2 + a l_2 m_2 C_1 C_2 + a l_2 m_3 C_1 C_2 \end{matrix} \right) \dot{\theta} + \left(\begin{matrix} l_3^2 m_3 C_{23}^2 + l_2^2 m_2 C_2^2 \\ + l_2^2 m_3 C_2^2 + 2l_2 l_3 m_3 C_{23} C_2 \end{matrix} \right) \dot{q}_1 \end{aligned} \quad (4.79)$$

$$\begin{aligned} \frac{dL}{d\dot{q}_2} = & -C_{\theta 1}(l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{x} - a \sin(q_1)(l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{\theta} \\ & + l_3 m_3 (l_3 + l_2 C_3) \dot{q}_3 - S_{\theta 1} \left(\begin{matrix} l_2 m_2 S_2 + l_2 m_3 S_2 \\ + l_3 m_3 S_{23} \end{matrix} \right) \dot{y} + \left(\begin{matrix} l_2^2 m_2 + l_2^2 m_3 \\ + l_3^2 m_3 + 2l_2 l_3 m_3 C_3 \end{matrix} \right) \dot{q}_2 \end{aligned} \quad (4.80)$$

$$\frac{dL}{d\dot{q}_3} = -l_3 m_3 C_{\theta 1} S_{23} \dot{x} - l_3 m_3 S_{23} S_{\theta 1} \dot{y} - a l_3 m_3 S_{23} S_1 \dot{\theta} + l_3 m_3 (l_3 + l_2 C_3) \dot{q}_2 + l_3^2 m_3 \dot{q}_3 \quad (4.81)$$

derivando con respecto al tiempo todas estas ecuaciones según el método de Euler – Lagrange se obtiene entonces:

$$\begin{aligned}
\frac{d}{dt} \left(\frac{dL}{d\dot{x}_p} \right) &= (m + m_2 + m_3) \ddot{x} - l_3 m_3 C_{\theta 1} S_{23} \ddot{q}_3 + \left(\begin{aligned} &-m_2 (S_{\theta} (a + l_2 C_1 C_2) + l_2 C_2 C_{\theta} S_1) \\ &-m_3 \left(\begin{aligned} &C_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ &+ S_{\theta} (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \end{aligned} \right) - am S_{\theta} \end{aligned} \right) \ddot{\theta} \\
&- S_{\theta 1} \left(\begin{aligned} &l_2 m_2 C_2 + l_2 m_3 C_2 \\ &+ l_3 m_3 C_{23} \end{aligned} \right) \ddot{q}_1 - C_{\theta 1} \left(\begin{aligned} &l_2 m_2 S_2 + l_2 m_3 S_2 \\ &+ l_3 m_3 S_{23} \end{aligned} \right) \ddot{q}_2 - 2l_3 m_3 C_{23} C_{\theta 1} \dot{q}_2 \dot{q}_3 + 2l_3 m_3 S_{23} S_{\theta 1} \dot{q}_1 \dot{q}_3 \\
&+ \left(\begin{aligned} &m_3 \left(\begin{aligned} &S_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ &- C_{\theta} (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \end{aligned} \right) \\ &- m_2 (C_{\theta} (a + l_2 C_1 C_2) - l_2 C_2 S_1 S_{\theta}) - am C_{\theta} \end{aligned} \right) \dot{\theta}^2 - C_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1^2 \\
&- C_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_2^2 - l_3 m_3 C_{23} C_{\theta 1} \dot{q}_3^2 - 2C_{\theta 1} (l_2 m_2 \cos(q_2) + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{\theta} \\
&+ 2S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_2 \dot{\theta} + 2l_3 m_3 S_{23} S_{\theta 1} \dot{q}_3 \dot{\theta} + 2S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_1 \dot{q}_2
\end{aligned} \tag{4.82}$$

$$\begin{aligned}
\frac{d}{dt} \left(\frac{dL}{d\dot{y}_p} \right) &= (m + m_2 + m_3) \ddot{y}_p + \left(\begin{aligned} &m_2 (C_{\theta} (a + l_2 C_1 C_2) - l_2 C_2 S_1 S_{\theta}) \\ &-m_3 \left(\begin{aligned} &S_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ &- C_{\theta} (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \end{aligned} \right) + am C_{\theta} \end{aligned} \right) \ddot{\theta} \\
&+ \left(\begin{aligned} &l_3 m_3 C_{23} C_{\theta 1} \\ &+ l_2 m_2 C_2 C_{\theta 1} + l_2 m_3 C_2 C_{\theta 1} \end{aligned} \right) \ddot{q}_1 - l_3 m_3 S_{23} S_{\theta 1} \ddot{q}_3 - S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \ddot{q}_2 \\
&- S_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1^2 - S_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_2^2 \\
&- \left(\begin{aligned} &m_2 (S_{\theta} (a + l_2 C_1 C_2) + l_2 C_2 C_{\theta} S_1) \\ &+ m_3 \left(\begin{aligned} &C_{\theta} (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ &+ S_{\theta} (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \end{aligned} \right) + am S_{\theta} \end{aligned} \right) \dot{\theta}^2 - (l_3 m_3 C_{23} S_{\theta 1}) \dot{q}_3^2 - 2l_3 m_3 C_{23} S_{\theta 1} \dot{q}_2 \dot{q}_3 \\
&- 2C_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_1 \dot{q}_2 - 2l_3 m_3 S_{23} C_{\theta 1} \dot{q}_1 \dot{q}_3 - 2l_3 m_3 S_{23} C_{\theta 1} \dot{q}_3 \dot{\theta} \\
&- (l_3 m_3 C_{23} S_{\theta 1} + l_2 m_2 C_2 S_{\theta 1} + l_2 m_3 C_2 S_{\theta 1}) 2\dot{q}_1 \dot{\theta} - \left(\begin{aligned} &l_3 m_3 S_{23} C_{\theta 1} + l_2 m_2 \sin(q_2) C_{\theta 1} \\ &+ l_2 m_3 S_2 C_{\theta 1} \end{aligned} \right) 2\dot{q}_2 \dot{\theta}
\end{aligned} \tag{4.83}$$

$$\begin{aligned}
\frac{d}{dt} \left(\frac{dL}{d\dot{\theta}_p} \right) &= \begin{pmatrix} -m_2(S_\theta(a+l_2C_1C_2)+l_2C_2C_\theta S_1) \\ -m_3(C_\theta(l_3C_{23}S_1+l_2C_2S_1) \\ +S_\theta(a+l_3C_{23}C_1+l_2C_1C_2))-amS_\theta \end{pmatrix} \ddot{x} + \begin{pmatrix} -m_2(C_\theta(a+l_2C_1C_2)-l_2C_2S_1S_\theta) \\ -m_3(S_\theta(l_3C_{23}S_1+l_2C_2S_1)+amC_\theta) \\ -C_\theta(a+l_3C_{23}C_1+l_2C_1C_2) \end{pmatrix} \ddot{y} \\
&+ \begin{pmatrix} m+m_2+m_3+a^2m+a^2m_2+a^2m_3+l_3^2m_3C_{23}^2 \\ +l_2^2m_2C_2^2+l_2^2m_3C_2^2+2al_3m_3C_{23}C_1 \\ +2l_2l_3m_3C_{23}C_2+2al_2m_2C_1C_2+2al_2m_3C_1C_2 \end{pmatrix} \ddot{\theta} + \begin{pmatrix} l_3^2m_3C_{23}^2+l_2^2m_2C_2^2+l_2^2m_3C_2^2 \\ +al_3m_3C_{23}C_1+2l_2l_3m_3C_{23}C_2 \\ +al_2m_2C_1C_2+al_2m_3C_1C_2 \end{pmatrix} \ddot{q}_1 \\
&-aS_1(l_2m_2S_2+l_2m_3S_2+l_3m_3S_{23})\ddot{q}_2 - al_3m_3S_{23}\sin(q_1)\ddot{q}_3 - aS_1(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_1^2 \\
&-aS_1 \begin{pmatrix} l_2m_2C_2+l_2m_3C_2 \\ +l_3m_3C_{23} \end{pmatrix} \dot{q}_2^2 - al_3m_3C_{23}S_1\dot{q}_3^2 + \begin{pmatrix} m_3(S_\theta(l_3C_{23}S_1+l_2C_2S_1)-amC_\theta) \\ -C_\theta(a+l_3C_{23}C_1+l_2C_1C_2) \\ -m_2(C_\theta(a+l_2C_1C_2)-l_2C_2S_1S_\theta) \end{pmatrix} \dot{\theta}\dot{x} \\
&-C_{1\theta}(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_1\dot{x} + S_{\theta 1}(l_2m_2S_2+l_2m_3S_2+l_3m_3S_{23})\dot{q}_2\dot{x} + (l_3m_3S_{23}S_{\theta 1})\dot{q}_3\dot{x} \\
&+ \begin{pmatrix} -m_2(S_\theta(a+l_2C_1C_2)+l_2C_2C_\theta S_1) \\ -m_3(C_\theta(l_3C_{23}S_1+l_2C_2S_1) \\ +S_\theta(a+l_3C_{23}C_1+l_2C_1C_2))-amS_\theta \end{pmatrix} \dot{\theta}\dot{y} - S_{\theta 1}(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_1\dot{y} \\
&-S_{\theta 1}(l_2m_2S_2+l_2m_3S_2+l_3m_3S_{23})\dot{q}_2\dot{y} - l_3m_3C_{\theta 1}S_{23}\dot{q}_3\dot{y} - 2aS_1(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_1\dot{\theta} \\
&+ \begin{pmatrix} -2l_2^2m_2C_2S_2-2l_2^2m_3C_2S_2-2l_3^2m_3C_{23}S_{23}-2al_3m_3S_{23}C_1-2l_2l_3m_3C_{23}S_2 \\ -2l_2l_3m_3S_{23}C_2-2al_2m_2C_1S_2-2al_2m_3C_1S_2 \end{pmatrix} \dot{q}_2\dot{\theta} \\
&-2l_3m_3S_{23}(l_3C_{23}+aC_1+l_2C_2)\dot{q}_3\dot{\theta} - aS_1(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_1\dot{q}_2 \\
&-aS_1(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_1\dot{q}_3 - aS_1(l_2m_2C_2+l_2m_3C_2+l_3m_3C_{23})\dot{q}_2\dot{q}_3
\end{aligned} \tag{4.84}$$

$$\begin{aligned}
\frac{d\left(\frac{dL}{d\dot{q}_1}\right)}{dt} = & -S_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\ddot{x} + C_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\ddot{y} \\
& + \left(l_3^2m_3C_{23}^2 + l_2^2m_2C_2^2 + l_2^2m_3C_2^2 + al_3m_3C_{23}C_1 \right) \ddot{\theta} + \left(l_3^2m_3C_{23}^2 + l_2^2m_2C_2^2 + l_2^2m_3C_2^2 \right. \\
& \left. + 2l_2l_3m_3C_{23}C_2 + al_2m_2C_1C_2 + al_2m_3C_1C_2 \right) \ddot{q}_1 \\
& + (-C_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23}))\dot{\theta}\dot{x} - C_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_1\dot{x} \\
& + (S_{1\theta}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23}))\dot{q}_2\dot{x} + (l_3m_3S_{23}S_{1\theta})\dot{q}_3\dot{x} - l_3m_3C_{1\theta}S_{23}\dot{q}_3\dot{y} \\
& - S_{1\theta}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{\theta}\dot{y} - S_{1\theta}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_1\dot{y} \\
& - C_{1\theta}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_2\dot{y} - a\sin(q_1)(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_1\dot{\theta} \\
& + \left(\begin{aligned} & -2l_2^2m_2C_2S_2 - 2l_2^2m_3C_2S_2 - 2l_2^2m_3C_{23}S_{23} \\ & -al_3m_3S_{23}C_1 - 2l_2l_3m_3C_{23}S_2 - al_2m_2C_1S \\ & -2l_2l_3m_3S_{23}C_{22} - al_2m_3C_1S_2 \end{aligned} \right) \dot{q}_2\dot{\theta} - l_3m_3S_{23}(2l_3C_{23} + a\cos(q_1) + 2l_2C_2)\dot{q}_3\dot{\theta} \\
& + (-l_3^2m_3\sin(2q_2 + 2q_3) - l_2^2m_2\sin(2q_2) - l_2^2m_3\sin(2q_2) - 2l_2l_3m_3\sin(2q_2 + q_3))\dot{q}_1\dot{q}_2 \\
& - 2l_3m_3S_{23}(l_3C_{23} + l_2C_2)\dot{q}_1\dot{q}_3
\end{aligned} \tag{4.85}$$

$$\begin{aligned}
\frac{d\left(\frac{dL}{d\dot{q}_2}\right)}{dt} = & -C_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\ddot{x} - S_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\ddot{y}.. \\
& .. - aS_1(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\ddot{\theta} + (l_2^2m_2 + l_2^2m_3 + l_3^2m_3 + 2l_2l_3m_3C_3)\ddot{q}_2.. \\
& .. + (l_3m_3(l_3 + l_2C_3))\ddot{q}_3 - l_2l_3m_3S_3\dot{q}_3^2 + (S_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23}))\dot{\theta}\dot{x}.. \\
& .. + (S_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23}))\dot{q}_1\dot{x} - C_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_2\dot{x}.. \\
& .. - l_3m_3C_{23}C_{\theta_1}\dot{q}_3\dot{x} - C_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\dot{\theta}\dot{y} - al_3m_3C_{23}\sin(q_1)\dot{q}_3\dot{\theta} \\
& .. - S_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_2\dot{y} - l_3m_3C_{23}S_{\theta_1}\dot{q}_3\dot{y} \\
& .. + (-aC_1(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23}))\dot{q}_1\dot{\theta} - 2l_2l_3m_3S_3\dot{q}_3\dot{q}_2 \\
& .. - C_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\dot{q}_1\dot{y} - aS_1(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{q}_2\dot{\theta}
\end{aligned} \tag{4.86}$$

$$\begin{aligned}
\frac{d\left(\frac{dL}{d\dot{q}_3}\right)}{dt} = & (-l_3m_3C_{\theta_1}S_{23})\ddot{x} - l_3m_3S_{\theta_1}S_{23}\ddot{y} - al_3m_3S_{23}S_1\ddot{\theta} + (l_3m_3(l_3 + l_2C_3))\ddot{q}_2.. \\
& .. + (l_3^2m_3)\ddot{q}_3 + (l_3m_3S_{23}S_{\theta_1})\dot{\theta}\dot{x} + (l_3m_3S_{23}S_{\theta_1})\dot{q}_1\dot{x} - l_3m_3C_{23}C_{\theta_1}\dot{q}_2\dot{x} - l_3m_3C_{23}C_{\theta_1}\dot{q}_3\dot{x} \\
& .. - l_3m_3C_{\theta_1}S_{23}\dot{\theta}\dot{y} - l_3m_3C_{\theta_1}S_{23}\dot{q}_1\dot{y} - al_3m_3C_{23}S_1\dot{q}_3\dot{\theta} - l_3m_3C_{23}S_{\theta_1}\dot{q}_2\dot{y} - l_3m_3C_{23}S_{\theta_1}\dot{q}_3\dot{y} \\
& .. - al_3m_3S_{23}C_1\dot{q}_1\dot{\theta} - al_3m_3C_{23}S_1\dot{q}_2\dot{\theta} - l_2l_3m_3S_3\dot{q}_2\dot{q}_3
\end{aligned} \tag{4.87}$$

derivando conforme Euler – Lagrange, respecto a posiciones lineales y angulares en función de

$$\frac{\partial L}{\partial x_p} = 0 \text{ y } \frac{\partial L}{\partial y} = 0 :$$

$$\begin{aligned}
\frac{dL}{d\theta_p} = & \left(m_3 \begin{pmatrix} S_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1) - .. \\ .. C_\theta (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \\ -m_2 (C_\theta (a + l_2 C_1 C_2) - l_2 C_2 S_1 S_\theta) - am C_\theta \end{pmatrix} \right) \dot{\theta} \dot{x} - C_{\theta 1} \begin{pmatrix} l_2 m_2 C_2 + l_2 m_3 C_2 \\ + l_3 m_3 C_{23} \end{pmatrix} \dot{q}_1 \dot{x} .. \\
& .. - \left(m_3 \begin{pmatrix} C_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1) + .. \\ .. S_\theta (a + l_3 C_{23} C_1 + l_2 C_1 C_2) \\ + m_2 (S_\theta (a + l_2 C_1 C_2) + l_2 C_2 C_\theta S_1) + am S_\theta \end{pmatrix} \right) \dot{\theta} \dot{y} + \begin{pmatrix} S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2) \\ + l_3 m_3 S_{23} \end{pmatrix} \dot{q}_2 \dot{x} .. \\
& .. + (l_3 m_3 S_{23} S_{\theta 1}) \dot{q}_3 \dot{x} - S_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{y} .. \\
& .. - C_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_2 \dot{y} - l_3 m_3 C_{\theta 1} S_{23} \dot{q}_3 \dot{y}
\end{aligned} \tag{4.88}$$

$$\begin{aligned}
\frac{dL}{dq_1} = & -a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{\theta}^2 - C_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{\theta} \dot{x} .. \\
& .. - C_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{x} + (S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23})) \dot{q}_2 \dot{x} .. \\
& .. - S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{\theta} \dot{y} - S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_1 \dot{y} .. \\
& .. - C_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_2 \dot{y} - a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{\theta} .. \\
& .. + l_3 m_3 S_{23} S_{\theta 1} \dot{q}_3 \dot{x} - a C_1 (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \dot{q}_2 \dot{\theta} - a l_3 m_3 S_{23} C_1 \dot{q}_3 \dot{\theta} - l_3 m_3 C_{\theta 1} S_{23} \dot{q}_3 \dot{y}
\end{aligned} \tag{4.89}$$

$$\begin{aligned}
\frac{dL}{dq_2} = & \begin{pmatrix} -l_2^2 m_2 C_2 S_2 - l_2^2 m_3 C_2 S_2 - l_3^2 m_3 C_{23} S_{23} \\ -a l_3 m_3 S_{23} C_1 - l_2 l_3 m_3 C_{23} S_2 \\ -l_2 l_3 m_3 S_{23} C_2 - a l_2 m_2 C_1 S_2 - a l_2 m_3 C_1 S_2 \end{pmatrix} \dot{\theta}^2 + \begin{pmatrix} S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2) \\ + l_3 m_3 S_{23} \end{pmatrix} \dot{\theta} \dot{x} .. \\
& .. + \left(\frac{-l_3^2 m_3 \sin(2q_2 + 2q_3) - l_2^2 m_2 \sin(2q_2) - l_2^2 m_3 \sin(2q_2) - 2l_2 l_3 m_3 \sin(2q_2 + q_3)}{2} \right) \dot{q}_1^2 \\
& .. + (S_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23})) \dot{q}_1 \dot{x} + (-C_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23})) \dot{q}_2 \dot{x} \\
& .. + (-l_3 m_3 C_{23} C_{\theta 1}) \dot{q}_3 \dot{x} + (-C_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23})) \dot{\theta} \dot{y} + (-l_3 m_3 C_{23} S_{\theta 1}) \dot{q}_3 \dot{y} \\
& .. + (-C_{\theta 1} (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23})) \dot{q}_1 \dot{y} + (-S_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23})) \dot{q}_2 \dot{y} \\
& .. + \begin{pmatrix} -2l_2^2 m_2 C_2 S_2 - 2l_2^2 m_3 C_2 S_2 - 2l_3^2 m_3 C_{23} S_{23} \\ -a l_3 m_3 S_{23} C_1 - 2l_2 l_3 m_3 C_{23} S_2 - 2l_2 l_3 m_3 S_{23} C_2 \\ -a l_2 m_2 C_1 \sin(q_2) - a l_2 m_3 C_1 \sin(q_2) \end{pmatrix} \dot{q}_1 \dot{\theta} - a l_3 m_3 C_{23} S_1 \dot{q}_3 \dot{\theta} \\
& .. - a S_{\theta 1} (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_2 \dot{\theta} - (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) g
\end{aligned} \tag{4.90}$$

$$\begin{aligned}
\frac{dL}{dq_3} = & (-l_3 m_3 S_{23} (l_3 C_{23} + a C_1 + l_2 C_2)) \dot{\theta}^2 + (-l_3 m_3 S_{23} (l_3 C_{23} + l_2 C_2)) \dot{q}_1^2 \\
& - l_2 l_3 m_3 S_3 \dot{q}_2^2 + (l_3 m_3 S_{23} S_{1\theta}) \dot{\theta} \dot{x} + (l_3 m_3 S_{23} S_{1\theta}) \dot{q}_1 \dot{x} - l_3 m_3 C_{23} C_{\theta 1} \dot{q}_2 \dot{x} \\
& - l_3 m_3 C_{23} C_{\theta 1} \dot{q}_3 \dot{x} - l_3 m_3 C_{\theta 1} S_{23} \dot{\theta} \dot{y} - l_3 m_3 C_{\theta 1} S_{23} \dot{q}_1 \dot{y} - l_3 m_3 C_{23} S_{1\theta} \dot{q}_2 \dot{y} \\
& - l_3 m_3 C_{23} S_{1\theta} \dot{q}_3 \dot{y} - a l_3 m_3 C_{23} S_1 \dot{q}_2 \dot{\theta} - l_3 m_3 S_{23} (2 l_3 C_{23} + a C_1 + 2 l_2 C_2) \dot{q}_1 \dot{\theta} \\
& + (-a l_3 m_3 C_{23} S_1) \dot{q}_3 \dot{\theta} + (-l_2 l_3 m_3 S_3) \dot{q}_2 \dot{q}_3 + (-l_3 m_3 C_{23}) g
\end{aligned} \tag{4.91}$$

Por definición $f_x = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x}$ donde $\frac{\partial L}{\partial x} \triangleright 0$ se encontrara la fuerza en x respecto a $\{R\}$ en el punto de interés; $f_x = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{x}} \right)$; tanto de la misma forma: $f_y = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{y}} \right)$ para encontrar la fuerza en y con respecto a $\{R\}$.

finalmente para encontrar el torque generado encima de la plataforma movil por: $\tau_\theta = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta}$

$$\begin{aligned}
\tau_\theta = & \begin{pmatrix} -m_2 (S_\theta (a + l_2 C_1 C_2) + l_2 C_2 C_\theta S_1) \\ -m_3 (C_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ + S_\theta (a + l_3 C_{23} C_1 + l_2 C_1 C_2)) - a m S_\theta \end{pmatrix} \ddot{x} + \begin{pmatrix} -m_2 (C_\theta (a + l_2 C_1 C_2) - l_2 C_2 S_1 S_\theta) \\ -m_3 (S_\theta (l_3 C_{23} S_1 + l_2 C_2 S_1) \\ - C_\theta (a + l_3 C_{23} C_1 + l_2 C_1 C_2)) + a m C_\theta \end{pmatrix} \ddot{y} \\
& + \begin{pmatrix} m + m_2 + m_3 + a^2 m + a^2 m_2 + a^2 m_3 + l_3^2 m_3 C_{23}^2 \\ + l_2^2 m_2 C_2^2 + l_2^2 m_3 C_2^2 + 2 a l_3 m_3 C_{23} C_1 \\ + 2 l_2 l_3 m_3 C_{23} C_2 + 2 a l_2 m_2 C_1 C_2 + 2 a l_2 m_3 C_1 C_2 \end{pmatrix} \ddot{\theta} + \begin{pmatrix} l_3^2 m_3 C_{23}^2 + l_2^2 m_2 C_2^2 + l_2^2 m_3 C_2^2 \\ + a l_3 m_3 C_{23} C_1 + 2 l_2 l_3 m_3 C_{23} C_2 \\ + a l_2 m_2 C_1 C_2 + a l_2 m_3 C_1 C_2 \end{pmatrix} \ddot{q}_1 \\
& - a S_1 (l_2 m_2 S_2 + l_2 m_3 S_2 + l_3 m_3 S_{23}) \ddot{q}_2 - a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1^2 \\
& - a S_1 \begin{pmatrix} l_2 m_2 C_2 + l_2 m_3 C_2 \\ + l_3 m_3 C_{23} \end{pmatrix} \dot{q}_2^2 - a l_3 m_3 C_{23} S_1 \dot{q}_3^2 - 2 a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{\theta} \\
& + \begin{pmatrix} -l_2^2 m_2 C_2 S_2 - l_2^2 m_3 C_2 S_2 - l_3^2 m_3 C_{23} S_{23} - a l_3 m_3 S_{23} C_1 \\ - l_2 l_3 m_3 C_{23} S_2 - l_2 l_3 m_3 S_{23} C_2 - a l_2 m_2 C_1 S_2 - a l_2 m_3 C_1 S_2 \end{pmatrix} 2 \dot{q}_2 \dot{\theta} - a l_3 m_3 S_{23} S_1 \ddot{q}_3 \\
& - 2 l_3 m_3 S_{23} (l_3 C_{23} + a C_1 + l_2 C_2) \dot{q}_3 \dot{\theta} - a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{q}_2 \\
& - a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_1 \dot{q}_3 - a S_1 (l_2 m_2 C_2 + l_2 m_3 C_2 + l_3 m_3 C_{23}) \dot{q}_2 \dot{q}_3
\end{aligned} \tag{4.92}$$

y en la misma forma $\tau_1 = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_1} \right) - \frac{\partial L}{\partial q_1}$

$$\begin{aligned}
\tau_1 = & -S_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\ddot{x} + C_{\theta_1}(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\ddot{y} \\
& + \left(l_3^2m_3C_{23}^2 + l_2^2m_2C_2^2 + l_2^2m_3C_2^2 + al_3m_3C_{23}C_1 \right) \ddot{\theta} + \left(l_3^2m_3C_{23}^2 + l_2^2m_2C_2^2 + l_2^2m_3C_2^2 \right) \ddot{q}_1 \\
& + \left(2l_2l_3m_3C_{23}C_2 + al_2m_2C_1C_2 + al_2m_3C_1C_2 \right) \ddot{q}_1 \\
& - \left(l_2^2m_2C_2S_2 + l_2^2m_3C_2S_2 + l_3^2m_3C_{23}S_{23} \right) 2\dot{q}_2\dot{\theta} - l_3m_3S_{23}(l_3C_{23} + l_2C_2)2\dot{q}_3\dot{\theta} \\
& + \left(l_2l_3m_3C_{23}S_2 + l_2l_3m_3S_{23}C_{22} \right) 2\dot{q}_2\dot{\theta} - l_3m_3S_{23}(l_3C_{23} + l_2C_2)2\dot{q}_3\dot{\theta} \\
& - \left(l_3^2m_3\sin(2q_2 + 2q_3) + l_2^2m_2\sin(2q_2) + l_2^2m_3\sin(2q_2) + 2l_2l_3m_3\sin(2q_2 + q_3) \right) \dot{q}_1\dot{q}_2 \\
& - 2l_3m_3S_{23}(l_3C_{23} + l_2C_2)\dot{q}_1\dot{q}_3 + aS_1(l_2m_2C_2 + l_2m_3C_2 + l_3m_3C_{23})\dot{\theta}^2
\end{aligned} \tag{4.93}$$

y que: $\tau_2 = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_2} \right) - \frac{\partial L}{\partial q_2}$

$$\begin{aligned}
\tau_2 = & -C_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\ddot{x} - S_{\theta_1}(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\ddot{y} \\
& .. - aS_1(l_2m_2S_2 + l_2m_3S_2 + l_3m_3S_{23})\ddot{\theta} + \left(l_2^2m_2 + l_2^2m_3 + l_3^2m_3 + 2l_2l_3m_3C_3 \right) \ddot{q}_2 \\
& + \left(\frac{l_3^2m_3\sin(2q_2 + 2q_3) + l_2^2m_2\sin(2q_2) + l_2^2m_3\sin(2q_2) + 2l_2l_3m_3\sin(2q_2 + q_3)}{2} \right) \dot{q}_1^2 \\
& .. + \left(l_3m_3(l_3 + l_2C_3) \right) \ddot{q}_3 - l_2l_3m_3S_3\dot{q}_3^2 + \left(\begin{aligned} & l_2^2m_2C_2S_2 + l_2^2m_3C_2S_2 + l_3^2m_3C_{23}S_{23} \\ & + al_3m_3S_{23}C_1 + l_2l_3m_3C_{23}S_2 \\ & + l_2l_3m_3S_{23}C_2 + al_2m_2C_1S_2 - al_2m_3C_1S_2 \end{aligned} \right) \dot{\theta}^2 \\
& .. + \left(\begin{aligned} & l_2^2m_2C_2S_2 + l_2^2m_3C_2S_2 + l_3^2m_3C_{23}S_{23} \\ & + l_2l_3m_3C_{23}S_2 + l_2l_3m_3S_{23}C_2 \end{aligned} \right) 2\dot{q}_1\dot{\theta} - 2l_2l_3m_3S_3\dot{q}_3\dot{q}_2 + \left(\begin{aligned} & l_2m_2C_2 + l_2m_3C_2 \\ & + l_3m_3C_{23} \end{aligned} \right) g
\end{aligned} \tag{4.94}$$

se considera además: $\tau_3 = \frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}_3} \right) - \frac{\partial L}{\partial q_3}$

$$\begin{aligned}
\tau_3 = & -l_3m_3C_{\theta_1}S_{23}\ddot{x} - l_3m_3S_{\theta_1}S_{23}\ddot{y} - al_3m_3S_{23}S_1\ddot{\theta} + \left(l_3m_3(l_3 + l_2C_3) \right) \ddot{q}_2 \\
& .. + l_3^2m_3\ddot{q}_3 + l_3m_3S_{23}(l_3C_{23} + aC_1 + l_2C_2)\dot{\theta}^2 + l_3m_3S_{23}(l_3C_{23} + l_2C_2)\dot{q}_1^2 \\
& .. + l_3m_3S_{23}(l_3C_{23} + l_2C_2)2\dot{q}_1\dot{\theta} + l_2l_3m_3S_3\dot{q}_2^2 + (l_3m_3C_{23})g
\end{aligned} \tag{4.95}$$

Igual que en el caso del robot anterior expresado de forma matricial, se obtiene el modelo dinámico del robot manipulador móvil:

$$f(t) = [fx \quad fy \quad \tau_0 \quad \tau_1 \quad \tau_2 \quad \tau_3]$$

$$\mathbf{f}(t) = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}(t) + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}(t) + \mathbf{g}(\mathbf{q})$$

donde, $\mathbf{q} = [x \quad y \quad \theta \quad q_1 \quad q_2 \quad q_3]$

Son estas las descripciones aproximadas de los modelos dinámicos de los cuatro tipos de robots utilizados en las aplicaciones de demostración y consecución de objetivos de este trabajo, también es necesario destacar que más de un robot de cada característica, se ha construido con orientación a las aplicaciones, de tal forma que han interactuado en los experimentos de la mayoría de las mencionadas aplicaciones, que son únicamente la comprobación de el cumplimiento de las tareas encomendadas a los diversos robots diseñados y construidos.

Capítulo V

5 DISEÑO E IMPLEMENTACIÓN DE LA RED INALÁMBRICA PARA COMUNICACIÓN ENTRE ESTACIONES ROBÓTICAS INTELIGENTES PARA TRABAJOS AUTÓNOMOS O COOPERATIVOS Y DE LA PLATAFORMA DE SUPERVISIÓN DEL SISTEMA DE ROBOTS MÓVILES.

5.1 Introducción.

Los robots en los entornos de operación actuales no actúan solos, es más, es muy poco probable encontrar una aplicación para ellos en la que sean entes completamente independientes e inteligentes, realizando una tarea en particular; antes, al contrario, estos agentes están en constante colaboración ya sea con una estación central, ya sea con más robots, o al final con un entorno de comunicaciones global.

El objetivo de esta tesis no solamente es el de construir una plataforma multisensorial que pueda hacer “algo”, sino el de desarrollar aplicaciones útiles, en las que pueda emplearse ésta plataforma; uno de los objetivos secundarios es implementar un sistema de colaboración, en el cual estos robots multisensoriales, y más del mismo o diferentes tipos de robots, se integren de tal forma que puedan juntos realizar tareas específicas, para lo cual el elemento más neurálgico es el de las comunicaciones.

Resulta obvio suponer que al ser dispositivos autónomos que ejecutan tareas de colaboración, necesitan de una red tipo Wireless, que les permita una movilidad adecuada dentro del área de cobertura de la red; es por eso que se analizaron las varias alternativas para emplear la mejor entre las existentes y disponibles para la comunicación entre todos los elementos que se involucran y que han sido anteriormente mencionados.

Hay muchos aspectos que se deben considerar al seleccionar una red inalámbrica, como: la rapidez de la comunicación entre ellos, la cobertura o alcance de la red, su topología, etc.; entonces la mejor solución no depende ni siquiera de las características tecnológicas, sino de la aplicación que debe presentar, y muchas veces una alternativa que abarque todos los requisitos necesarios. Por eso se analizan a continuación las alternativas inalámbricas disponibles, con el fin de seleccionar a la mejor.

5.2 Sistemas por ultrasonidos.

Cuyos componentes son normalmente los sensores ultrasónicos que usan ondas sonoras de altas frecuencias, de tal forma que se tornan imperceptibles a los oídos humanos, es decir usan tonos no audibles; por encima de 15 – 20 KHz. Consisten básicamente en generar una serie de tonos por encima de las frecuencias audibles de los seres humanos en los altavoces de los dispositivos

transmisores, en los que se codifica la información a transmitir. Posteriormente estas ondas sonoras son captadas por el micrófono del terminal receptor y decodificadas hasta obtener los datos útiles. El ancho de banda es muy reducido y además requiere usar altavoces y micrófonos que necesariamente deben incorporarse; Para los humanos esto no suele ser un problema, ya que la mayoría de los adultos, no pueden percibir más allá de los 15-16 KHz, aunque los niños si pueden llegar a percibir esos 20 KHz y por supuesto las mascotas sí que pueden oír por encima de estas frecuencias, con lo que podría darse el caso de que les moleste cada vez que se envíen datos. Muchas veces este tipo de sensores ultrasónicos requieren un determinado tiempo de espera en operación hasta que las vibraciones en el sensor desaparezcan y volver a operar; Los sistemas ultrasónicos son empleados generalmente en sistemas de detección para robots, y muy pocas veces en el envío de datos entre dispositivos como elemento de transmisión de información.

5.3 Sistemas Infrarrojos.

Utilizan un sistema de luz con una longitud de onda más grande que la luz visible, esto es de entre 700nm y 1 mm. Permiten la comunicación entre nodos, para lo cual se basan en el uso de un conjunto de leds; cuando trabajan en modo punto a punto, los patrones de radiación del emisor y del receptor deben de estar lo más cerca posible y que su alineación sea correcta; como resultado, el modo punto a punto requiere una línea de visión entre las dos estaciones a comunicarse. En su modo de operación cuasi-difuso, usan métodos de emisión radial, es decir que cuando una estación emite una señal óptica, esta puede ser recibida por todas las estaciones al mismo tiempo en la célula, en este modo las estaciones se comunican entre sí por medio de superficies reflectantes; no es necesaria la línea de visión entre dos estaciones, pero sí deben de estarlo con la superficie de reflexión. Además, es recomendable que las estaciones estén cerca de la superficie de reflexión, esta puede ser pasiva o activa. En las células basadas en reflexión pasiva, el reflector debe de tener altas propiedades reflectivas y dispersivas, mientras que en las basadas en reflexión activa se requiere de un dispositivo de salida reflexivo, conocido como satélite, que amplifica la señal óptica. La reflexión pasiva requiere más energía, por parte de las estaciones, pero es más flexible de usar. Este tipo de sensores a pesar de tener un comportamiento no lineal, existen algunos sensores que se basan en desplazamiento de fase y con ellos logran una cobertura mayor de hasta 10 metros.

Lamentablemente este tipo de sensores y elementos de comunicación, funcionan bien en distancias pequeñas, pero al necesitar el mismo campo visual del transmisor y el receptor, no como se requiere a distancias más largas.

Las comunicaciones IR pueden ser configuradas de dos formas, en función del grado de direccionalidad del transmisor y receptor y la necesaria línea de vista, es por esta razón que se pueden configurar enlaces dirigidos y no dirigidos:

- Los enlaces dirigidos usan equipos de transmisión y recepción de alta direccionalidad, que apuntaran los haces entre ellos o hacia un punto común para establecer los enlaces, normalmente se usan en configuraciones punto a punto.
- Los enlaces no dirigidos emplean elementos de gran amplitud, reduciendo de esta forma la necesidad de la línea de visión y logrando así, mayor cobertura, se emplean más en configuraciones multi punto.

5.4 Sistemas FSO.

Corresponden a sistemas de tecnología Free Space Optics, tecnología que transmite señales ópticas en el espacio libre, de forma que se parecen a las señales que viajan por las fibras ópticas, pero son transmitidas libremente al ambiente. Usan un diodo LED o uno LASER como elemento de transmisión, por el cual se envía la información digital convertida en luz, se usa un procedimiento denominado modulación en potencia, que modula señales eléctricas en AM.

Pueden realizar conexiones entre redes LAN en distancias de cobertura pequeñas (tipo campus), por eso su campo de aplicación ha sido diverso; cuentan con una velocidad aproximada de entre 100 y 2500 Mbps, la calidad del enlace es altamente dependiente de los factores atmosféricos como lluvia, niebla, polvo y calor. En el espacio exterior, el alcance de las comunicaciones ópticas de espacio libre en la actualidad es del orden de varios miles de kilómetros, pero tiene el potencial de alcanzar distancias interplanetarias de millones de kilómetros, utilizando telescopios ópticos como expansores de haz.

5.5 Sistemas de Radio.

Aquí se presentan las principales técnicas de comunicación por radiofrecuencia, entre ellos Bluetooth, ZigBee y Wifi.

Está basada en la transmisión de ondas radioeléctricas del espectro electromagnético en una banda que va desde los 3 Hz hasta los 300 GHz. Cada uno de ellos (son muchos y de diversos fabricantes), tienen una orientación específica, así que, mientras Wifi se orienta a comunicaciones de larga distancia, ZigBee, lo hace para distancias cortas entre equipos de control de red y domótica; por otro lado Bluetooth orienta su aplicación a comunicación entre dispositivos móviles, cada una de ellas presenta restricciones con respecto a algunos aspectos tecnológicos y de ambiente, así como también ventajas y características de orientación especiales.

En la Figura 5.1 se observa la distribución del espectro radioeléctrico por frecuencias contra la distancia de cobertura que poseen cada una de ellas, ahí se puede definir también la orientación a unas aplicaciones en específico de cada una. Se analizan sus características fundamentales, de tal suerte que se pueda discriminar una de otra para utilizar la que más convenga en la robótica que se ha empleado.

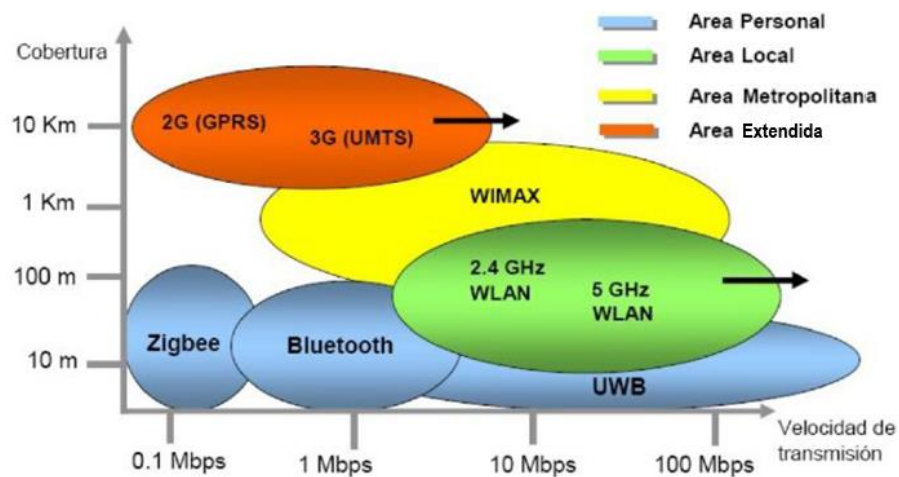


Figura 5.1. Espectro de aplicación de redes inalámbricas y cobertura.

5.5.1 Sistema Bluetooth.

Es una especificación industrial para redes inalámbricas de área personal (WPAN) creado por Bluetooth Special Interest Group, Inc. que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz. Frecuentemente se la usa para la transmisión de datos entre teléfonos móviles, dispositivos electrónicos, juguetes, etc.

Una de sus principales ventajas, al contrario de lo que se analizaba en los párrafos anteriores, no se necesita línea de vista ni cosa parecida para la comunicación entre dispositivos, además que permite transmisión de voz y datos por su portadora de 2.4 GHz. Se basa en el estándar IEEE 802.15.1; y se clasifican en tres clases en virtud a su potencia y alcance de acuerdo con:

Tabla 5.1. Clases de dispositivos Bluetooth

Dispositivo Bluetooth	Potencia Máxima	Alcance (m)
Clase 1	100 mW (20dBm)	100
Clase 2	2.5 mW (4dBm)	10 – 15
Clase 3	1 mW (0 dBm)	1

En donde la clase 1 es utilizada de forma prominente por el sector industrial, la clase 2 la usan la mayoría de los dispositivos móviles; sin embargo, el alcance efectivo tendrá variaciones debidas a la cobertura, las condiciones de propagación, las características de las antenas y el estado de la batería, así como la variación personalizada de su radio de alcance.

A continuación, se reproduce la evolución de la norma al respecto de los años y sus características:

- Bluetooth v1.0 y v1.0B. En 1999 se lanzaron las primeras versiones de Bluetooth. Hoy en día ya están prácticamente obsoletas debido a que dieron bastantes fallos de comunicación entre dispositivos de diferentes compañías. Aunque el gran fallo era que cada vez que se quería enviar datos, en cada transmisión se enviaban también la dirección privada propia del dispositivo, lo que hacía perder el anonimato.
- Bluetooth v1.1. En esta nueva versión, se corrigieron los errores de las versiones precedentes. Aspectos a destacar son la utilización del estándar IEEE 802.15.1-2002 y la incorporación de un indicador de fuerza de señal recibida, RSSI.
- Bluetooth v1.2. La conexión en esta versión es mucho más rápida (Ancho de banda de 1 Mbit/s). Introduce la técnica de detección de otros dispositivos Bluetooth en un radio de alcance.
- Bluetooth v2.0 + EDR. Compatible con la versión anterior, la principal diferencia radica en la introducción de una velocidad de datos mejorada (EDR) para acelerar precisamente esa transferencia de datos. La tasa nominal de EDR es de 3 Mbit/s, aunque en la práctica sea de 2,1 Mbit/s.
- Bluetooth v2.1 + EDR. Las principales características de esta versión son: por un lado, la mejora del funcionamiento de emparejamiento entre dispositivos Bluetooth y por el otro, una mayor seguridad de los propios datos añadiendo Secure Simple Pairing (SSP), que permite un mejor filtrado de los datos y una seguridad superior a la de la versión anterior.
- Bluetooth v3.0 + HS. Soporta velocidades de transferencia de datos de hasta 24 Mbit/s aunque no con el propio enlace Bluetooth. La principal novedad es la incorporación del estándar 802.11 para el envío y recepción de grandes paquetes de datos a una alta velocidad (High Speed) que, por lo general, está relacionado con Wifi.
- Bluetooth v4.0. Lanzado en 2010 y conocido también como Bluetooth Low Energy, se caracteriza precisamente por eso, por consumir muy poca energía. Se empieza a integrar en dispositivos inalámbricos como relojes digitales.
- Bluetooth v4.1. Esta versión, se corrigen las interferencias con 4G que pudiera haber, y se introduce la conexión simultánea en la que un mismo dispositivo se puede usar de emisor y receptor de datos al mismo tiempo. Esta versión apenas cambia el hardware así que es perfectamente compatible con la versión anterior.

- Bluetooth v4.2. Lanzada a finales de 2013, es la versión de Bluetooth que incorpora la posibilidad de conectarse a Internet sin necesidad de Wi-Fi. Introduce, además, un nuevo cifrado FIPS para asegurar la información. Es la versión que se incorporan en los últimos Smartphones de su época tales como el nuevo iPhone 6s o el Samsung Galaxy S7.
- Bluetooth v5.0. a finales de 2016, Bluetooth SIG, ha mejorado el rendimiento de la versión anterior, posibilitando una transferencia de datos dos veces más rápida, y un ancho de banda ocho veces mayor. Es decir, se puede transferir datos ocho veces más grandes el doble rápido; además, el Bluetooth 5.0 aumenta el rango de distancia de la señal en cuatro veces, por lo que se puede separar más de los accesorios cuando estos estén conectados.
- Bluetooth v5.1 se ha presentado a finales de octubre de 2019, Todas las novedades de esta nueva versión de Bluetooth giran en torno a la localización. Los dispositivos con Bluetooth 5.1 podrán saber la ubicación de otros dispositivos a los que estén conectados. Esta detección no será 100% precisa como el caso del GPS, pero sí podrá determinar una ubicación con un margen de unos cuantos centímetros. Teniendo en cuenta que la versión actual de Bluetooth detecta otros dispositivos con un margen de entre 1 y 10 metros, haber reducido esa distancia a apenas unos centímetros es todo un logro en este sentido. Además, un dispositivo también podrá identificar la dirección de donde proviene una señal que está buscando. Este tipo de tecnología a detalle también ayudaría en entornos industriales. Con ella, se podrán colocar receptores Bluetooth a los productos que haya ubicados en un gran almacén, y con un dispositivo Bluetooth 5.1 podrán buscarse y saber en qué parte del almacén puedes encontrar cada producto.

Abordando el concepto de robótica de colaboración, se puede mencionar que Bluetooth no está ausente, y eso se debe en gran parte a su capacidad de enlazar conexiones multipunto de nodos en total, haciendo atractivo el análisis de eficiencia para el uso en aplicaciones como estas; la implementación de este tipo de redes o redes híbridas, está basada también en el uso dentro de un robot con plataformas multi-comunicación que permitan mezclar las señales entre ellas, esto en la práctica se puede hacer con el uso de tarjetas embebidas de desarrollo montadas sobre cada robot o una estación.

5.5.2 Sistemas WiFi.

Es otro sistema de comunicaciones inalámbrico altamente difundido y utilizado, en la actualidad especialmente por el grandioso avance de los dispositivos móviles inteligentes, como los smartphones y las Tablet PC, etc. Es diseñado principalmente para la aplicación de redes de área local (WLAN), que permite la interconexión con múltiples dispositivos para intercambiar datos entre sí; usa de igual forma ondas radioeléctricas, que permiten conexiones extremadamente rápidas y seguras, pues al ser una red de tipo LAN, implícitamente acarrea todos sus sistemas de seguridad, asegurando un alcance de entre 90 y 100 metros.

Se compone de tres elementos fundamentales que son: las estaciones, el Access point y el medio de transmisión; tiene la finalidad de compartir datos con las estaciones conectadas a su misma red y a otras redes conectadas a ella. Está basado en el estándar IEEE 802.11 que usa el espacio libre para la transmisión de datos, siempre y cuando exista un punto de enlace entre las estaciones y la red (Access point).

Usa las dos primeras capas del modelo OSI, es decir la capa física y la capa de enlace; la primera, define las especificaciones en cuanto al tipo de modulación que emplea en las portadoras, o la frecuencia de operación, la segunda en cambio definirá la forma de establecer la conexión, elegir el camino para la entrega de los datos.

Al igual que la especificación anterior ha tenido una cierta evolución a través de todos estos años:

- 802.11 legacy. Es la primera versión original de Wi-Fi que se publicó en 1997. Con velocidades de 1 y 2 Mbps, las señales se transmitían vía infrarrojos.
- 802.11a. Un par de años más tarde se aprobó este nuevo estándar. El 802.11a que trabaja en la banda de frecuencias de 5 GHz y aunque sus velocidades teóricas puedan llegar a 54 Mbps, tiene velocidades reales en torno a los 20 Mbps. Es capaz de soportar 64 usuarios por cada punto de acceso con 64 portadoras OFDM.
- 802.11b. El estándar 802.11b, sin embargo, trabaja en la banda de 2.4 GHz con velocidades de 11 Mbps, que en la realidad están en torno a los 6 - 7 Mbps. Utilizando modulaciones de espectro ensanchado de tipo DSSS, es el tipo de tecnología Wi-Fi más extendida.
- 802.11g. La incompatibilidad de los sistemas anteriores lleva en 2003 a la creación de este estándar, que al igual que el 802.11b trabaja en la banda de 2.4 GHz, aunque alcanzando velocidades de 54 Mbps (22 Mbps reales). Utiliza tecnología OFDM, pero también es capaz de implementar la tecnología DSSS del estándar 802.11b. Es compatible con el estándar 802.11b, pero no con el 802.11a. Sin embargo, el uso combinado de los estándares 802.11b y 802.11g en dos dispositivos distintos, provoca que la velocidad de transmisión se reduzca a la del de menor tasa binaria.
- 802.11n. Cinco años más tarde, en septiembre de 2009 se creó este nuevo estándar. La principal diferencia radica en que puede trabajar en las bandas de 2.4 GHz y 5 GHz simultáneamente, por lo que es compatible con cualquier dispositivo basado en versiones precedentes de Wi-Fi. Aumenta el límite teórico a 600 Mbps, aunque ronda los 80-100 Mbps. Incorpora además la tecnología MIMO (Multiple Input Multiple Output) permitiendo utilizar varios canales a la vez para el envío y recepción de datos.

- 802.11ac. Es una mejora del estándar anterior lanzada al mercado en enero de 2014. Mejora la tasa de transferencia teórica de 1.3Gbps y reales de hasta los 433 Mbps. Opera en la banda de 5GHz evitando así interferencias con otras redes Wi-Fi y además introduce el denominado Beamforming capaz de focalizar la señal de radio, reconociendo elementos que puedan bajar el rendimiento (muros, paredes...) y evitándolos en la manera de lo posible. Actualmente es el estándar que incorporan la mayoría de los nuevos smartphones de las grandes compañías.
- 802.11ad. Es el estándar más reciente. Al ser multibanda, es capaz de llegar a una velocidad de 800 Mbps en 2.4 GHz, 1.7 Gbps en 5 GHz y de 4.5Gbps en 60 GHz. Fue aprobado a finales de 2014, pero no ha sido hasta este año, en enero de 2016 ha sido introducido al mercado. El primer router del mundo con 802.11ad lo ha lanzado TP-Link con Talón AD7200. Por su parte, la marca Acer ha hecho lo mismo con los portátiles, siendo TravelMate P648 el primero del mundo. En cuanto a los smartphones, la firma israelí Sirin Labs ha desarrollado Solarin, el primer smartphone con ese estándar, siendo, además, el smartphone más caro del mundo.
- 802.11ah y 802.11ax. Son los próximos estándares que lanzará WIFI Alliance. Lo que se sabe por el momento es que 802.11ah podrá operar en la banda de 900Mhz y que 802.11ax pretende llegar a velocidades de 10Gbps.
- IEEE 802.11ax, comercializado como Wi-Fi 6 por Wi-Fi Alliance, es uno de los dos estándares de especificaciones de Wi-Fi de IEEE 802.11 que espera una implementación completa a fines de 2019; el otro es IEEE 802.11ay. Pueden considerarse como inalámbricos de alta eficiencia. 802.11ax está diseñado para operar en todas las bandas ISM entre 1 y 6 GHz cuando estén disponibles para uso 802.11, además de las bandas de 2.4 y 5 GHz ya asignadas. Los dispositivos presentados en CES 2018 reclamaron una combinación de 11 Gbit/s de velocidades de datos teóricas. Para implementaciones densas, las velocidades de rendimiento son 4 veces más altas que IEEE 802.11ac, aunque la velocidad de datos nominal es solo un 37% más rápida como máximo. La latencia también ha bajado un 75%. Para mejorar la utilización eficiente del espectro, la nueva versión introduce mejores métodos de control de potencia para evitar interferencias con redes vecinas, acceso múltiple por división de frecuencia ortogonal (OFDMA), 1024-QAM de orden superior y dirección de enlace ascendente agregada con el enlace descendente de MIMO y MU-MIMO para aumentar aún más el rendimiento, así como mejoras en la confiabilidad del consumo de energía y protocolos de seguridad como Target Wake Time y WPA3.

La gran diferencia entre WiFi y los demás sistemas de comunicaciones por radio es su gran capacidad de canal, es decir la gran cantidad de datos que se pueden transferir a altas

velocidades. La gran aplicación de esta tecnología en robots colaborativos le hace una de las más atractivas tecnologías de aplicación, a pesar de que su latencia también es considerable en aplicaciones de control simultáneo de robots móviles especialmente.

5.5.3 Sistema ZigBee.

Basado en el protocolo 802.15.4, es un estándar de comunicaciones inalámbricas para redes de área personal, fue creado por IEEE para ser utilizado en una amplia variedad de aplicaciones que requieran comunicaciones inalámbricas sencillas, de corto alcance y potencia limitada. La norma 802.15.4 es un facilitador de aplicaciones, en la cual el valor no está en la capacidad inalámbrica sino en las aplicaciones.

Zigbee es una tecnología inalámbrica de baja tasa de transferencia, bajo consumo de energía. La IEEE y Zigbee Alliance han estado trabajando estrechamente para especificar toda la pila de protocolo. Este protocolo sigue la definición por capas del modelo de referencia de interconexión de sistemas abiertos OSI (Open System Interconnection). IEEE 802.15.4, se centra en la especificación de las dos capas inferiores del protocolo (capa física PHL, control de acceso al medio MAC). Por otro lado, Zigbee Alliance, proporciona las capas superiores de la pila del protocolo desde la capa de red hasta la de aplicación.

Las Red Inalámbrica de Área Personal (WPAN- Wireless Personal Area Network, se centran en un espacio alrededor de una persona u objeto que normalmente se extiende hasta 10m en todas las direcciones. El foco de WPAN es de bajo costo, de baja potencia, de corto alcance y de tamaño muy pequeño.

El grupo de trabajo IEEE 802.15 se forma para crear la norma WPAN. Este grupo actualmente ha definido tres clases de WPAN que se diferencian por la velocidad de datos, la descarga de la batería y la calidad de servicio (QoS). La alta velocidad de datos WPAN (IEEE 802.15.3) es adecuado para aplicaciones multimedia que requieren muy altas QoS. Tasa Media WPAN (IEEE 802.15.1 / bluetooth) se encargará de una variedad de tareas que van desde teléfonos celulares para comunicaciones PDA y tienen QoS adecuados para las comunicaciones de voz. Las principales características del estándar IEEE 802.15.4 WPAN es la flexibilidad de la red, de bajo costo, muy bajo consumo de energía, y baja velocidad de datos en unas redes móviles ad hoc permitiendo la conexión de diversos dispositivos económicos fijos, portátiles y móviles.

Los dispositivos ZigBee se clasifican de acuerdo a su funcionalidad en:

Dispositivos de funcionalidad completa, conocidos como FFD o nodos activos por su capacidad de cálculo y operación; incluye un módulo general de comunicaciones por el cual se establece el intercambio de datos con un RFD (dispositivos de funcionalidad reducida) u otro FFD; los modos de operación varían entre coordinador, router y simplemente dispositivo final.

Dispositivos de funcionalidad reducida, conocidos como nodos pasivos que no pueden comunicarse entre ellos, sino a través de un FFD, de forma general estos actúan como sensores de la red.

En redes ZigBee encontramos la estructura de tres tipos de elementos que son:

- **Coordinador ZigBee**; que son los elementos de mayor nivel y robustez de la red, solamente puede existir un coordinador por cada red ZigBee, y tiene como función principal iniciar la operación de la red y controlarla; es el que asigna las direcciones de red. Además de que al igual que los demás routers será el encargado de encaminar los mensajes a los diferentes usuarios dispositivos de la red, son FFD.
- **Router ZigBee**, son componentes que no necesariamente deben estar presentes en una red, tienen como funcionalidad la de conectar dispositivos separados enrutando mensajes para que lleguen al destino, se usan también como expansores de la red y creadores de nuevas rutas de conexión, también son FFD.
- **Dispositivo final ZigBee**, son nodos que solamente se limitan a comunicarse con los nodos padres ya sea este router o coordinador, son RFD y no participan en tareas de transmisión ni control de la red.

A diferencia de Bluetooth y WiFi, con ZigBee es posible implementar redes de control y automatización usando cuatro topologías de red distintas: estrella, entre pares, árbol de grupos y malla.

- Topología en estrella.** En la topología en estrella, se establece la comunicación entre los dispositivos y un controlador central único, llamado el coordinador de PAN. El coordinador de PAN puede ser conectado al fluido eléctrico, mientras que los otros dispositivos será probable que sean alimentados por baterías. Las aplicaciones que se benefician de esta topología incluyen domótica, computadoras personales (PC's) periféricos, juguetes y juegos. Después de un FFD se activa por primera vez, puede establecer su propia red y convertirse en el coordinador de PAN. Cada red de inicio elige un identificador de PAN, que no está utilizado actualmente por cualquier otra red dentro de la esfera de radio de influencia. Esto permite que cada red en estrella para operar de forma independiente.
- Topologías entre pares.** En topología de punto a punto, también hay un coordinador de PAN. En contraste con topología en estrella, cualquier dispositivo puede comunicarse con cualquier otro dispositivo, siempre y cuando están en el rango de uno al otro. Una red peer-to-peer puede ser ad hoc, de auto-organización y auto-recuperación. Las aplicaciones como el control industrial y la vigilancia, las redes de sensores inalámbricos de activos y seguimiento de inventario se beneficiarían de una topología de este tipo. También permite

múltiples saltos para encaminar mensajes desde cualquier dispositivo a cualquier otro dispositivo en la red, proporcionando fiabilidad por enrutamiento múltiples.

- c. **Topología árbol de grupos.** Esta denominación corresponde a un híbrido que se forma aplicando una estructura de árbol a varias estrellas. Cada estrella es un grupo (clúster), y el conjunto será un árbol de grupo (cluster tree). Red de árbol de grupo es un caso especial de una red peer-to-peer en la que la mayoría de los dispositivos son FFDs y un RFD pueden conectarse a una red de árbol de grupos como un nodo único al final de cada rama. Cualquiera de los FFD puede actuar como coordinador y proporcionar servicios de sincronización con otros dispositivos y coordinadores. Sin embargo, sólo uno de estos coordinadores es el coordinador del PAN.

El coordinador del PAN se forma con el primer grupo mediante el establecimiento de sí mismo como la cabeza de grupo (CLH - Cluster head) con un identificador de grupo (CID - cluster identifier) de cero, elegir un identificador PAN sin usar y transmitir tramas de señalización a los dispositivos vecinos. Un dispositivo candidato que reciba una trama de baliza puede solicitar unirse a la red en la CLH. Si el coordinador PAN permite unirse al dispositivo, añadirá este nuevo dispositivo como un dispositivo secundario en su lista de vecinos. El dispositivo recién unido añadirá la CLH como su padre en su lista de vecinos y empezará a transmitir balizas periódicas de tal manera que otros dispositivos candidatos pueden entonces unirse a la red en ese dispositivo. Una vez que se cumplen los requisitos de aplicación o de la red, el coordinador PAN puede instruir a un dispositivo para convertirse en el CLH de un nuevo grupo adyacente a la primera. La ventaja de esta estructura en clúster es el aumento del área de cobertura a costa de una mayor latencia de mensajes.

- d. **Topología Malla (mesh).** La estructura de la topología de malla es similar a la de la topología de árbol, con el coordinador en la parte superior de una estructura en forma de árbol. El coordinador está vinculado a un conjunto de routers y dispositivos finales, sus hijos. Un router puede entonces estar vinculado a más routers y dispositivos finales, sus hijos. Esto puede continuar a un número de niveles. Sin embargo, las reglas de comunicación son más flexibles que en los nodos de router dentro del alcance del otro puede comunicarse directamente. La topología de malla da lugar a la propagación de mensajes más eficiente, y significa que posee rutas alternativas en donde se pueden encontrar si un enlace falla o hay congestión. Una función de "descubrimiento de ruta" es siempre lo que permite a la red para encontrar la mejor ruta disponible para un mensaje.

5.5.4 Sistemas de Telefonía móvil.

En la actualidad con la incorporación en los mercados cada vez más creciente de teléfonos inteligentes, la telefonía se convirtió en una red de transmisiones de datos de todo tipo, siendo

una parte o un servicio mínimo el de transferir datos con mensajes sonoros (objetivo de la telefonía); más aún la expansión agresiva del internet móvil, hizo que los datos en esta red sean transferidos de forma rápida y segura a todos los destinatarios, en forma de datos, texto, imágenes, sonidos, voz, etc.; de estos existen incluso sofisticados sistemas de control de seguridad e industrial que permiten supervisar y operar en tiempo real tomado acciones correctivas inmediatas.

Los sistemas existentes y ampliamente difundidos son:

- a. **Sistemas GSM**, procede de la abreviatura: Sistemas Globales para las Comunicaciones Móviles, y es un estándar mundial en telefonía móvil, que permitía unificar el servicio telefónico de tal suerte que se pueda desde un teléfono móvil comunicarse con cualquier parte del mundo.

Utiliza la banda de frecuencia de 900 MHz, pero hay en total 14 bandas más definidas, con sus respectivas asignaciones de canal, en el estándar técnico TS 45.0051 del consorcio 3GPP, que reemplaza al TS 05.05; de acuerdo a la Tabla 5.2.

Tabla 5.2. Principales bandas de frecuencia en GSM

SISTEMA	BANDA	FRECUENCIA		ASIGNACION DEL CANAL
		Subida	bajada	
T-GSM-380	380	380.2–389.8	390.2–399.8	dinámica
T-GSM-410	410	410.2–419.8	420.2–429.8	dinámica
GSM-450	450	450.4–457.6	460.4–467.6	259–293
GSM-480	480	478.8–486.0	488.8–496.0	306–340
GSM-710	710	698.0–716.0	728.0–746.0	dinámica
GSM-750	750	747.0–762.0	777.0–792.0	438–511
T-GSM-810	810	806.0–821.0	851.0–866.0	dinámica
GSM-850	850	824.0–849.0	869.0–894.0	128–251
P-GSM-900	900	890.2–914.8	935.2–959.8	1–124
E-GSM-900	900	880.0–914.8	925.0–959.8	975–1023, 0-124
R-GSM-900	900	876.0–914.8	921.0–959.8	955–1023, 0-124
T-GSM-900	900	870.4–876.0	915.4–921.0	dinámica
DCS-1800	1800	1710.2–1784.8	1805.2–1879.8	512–885
PCS-1900	1900	1850.0–1910.0	1930.0–1990.0	512–810
T-GSM-380	380	380.2–389.8	390.2–399.8	dinámica

Se le conoce también como 2G o estándar de segunda generación, pues en la en esta generación los canales de voz completamente digitales.

- b. Sistemas GPRS**, es la evolución de GSM en el sentido en que ofrece una mejor transmisión de datos a una rapidez que va desde los 56 Kbps hasta los 114 Kbps, mucho más rápido que GSM (9,6 Kbps), su novedad más importantes es el paso de la conmutación de circuitos a la conmutación de paquetes, permite hacer llamadas telefónicas al mismo tiempo que se están transfiriendo datos, pues para acceder a datos usa el protocolo IP, mientras que para la transmisión de las señales de voz sigue utilizando GSM, al ser la velocidad de los datos mayores que las de la voz, se permite un rango más grande de movilidad.
- c. Sistemas 3G, 4G y 5G**, Constituyen las evoluciones de GPRS ofrecen (3G) mejoras en las antenas, conexiones más fiables, mayor calidad de la voz y sobre todo mayor velocidad en la transferencia de datos que van desde los 384 Kbps hasta los 2 Mbps, haciendo posible que aparezcan más aplicaciones en tiempo real, como video y sonido, aumentando la seguridad en la red y los sistemas de autenticación. Con el paso del tiempo surgió la tecnología HSPA que mejora muy claramente las tasas de transferencia, llegando a los límites de los 14 Mbps hasta los 84 Mbps.

Por su parte, 4G, es el estándar de comunicaciones móviles más moderno que hay en la actualidad. Está basado completamente en el protocolo IP. Utiliza la tecnología LTE (Long Term Evolution) y como en 3G, su principal característica es proveer de velocidades de transferencia altas, en este caso, se pueden alcanzar alrededor de los 300Mbps (bajada) y los 86Mbps (subida). Es por tanto fácil imaginar, que, con tal tasa de transferencia de datos, el abanico de posibilidades de aplicaciones en el mundo de la robótica es bastante amplio.

5G son las siglas utilizadas para referirse a la quinta generación de tecnologías de telefonía móvil. Es la sucesora de la tecnología 4G. Actualmente está disponible su primera versión estandarizada (Release 15 - Stand Alone) aunque las empresas de telecomunicaciones continúan investigando nuevas tecnologías para posteriores versiones. Aunque a 2019 se lanzaron las primeras redes comerciales, se prevé que su uso se extienda exponencialmente desde 2020.

Más allá de la mejora de la velocidad, se espera que la 5G desate un ecosistema masivo de IoT (Internet de las cosas) donde las redes pueden satisfacer las necesidades de comunicación de miles de millones de dispositivos conectados, con las compensaciones correctas entre velocidad, latencia y costo. Sus características principales son:

- Una tasa de transferencia de datos de hasta 10Gbps; de 10 a 100 veces mejor que las redes 4G y 4.5G
- Latencia de 1 milisegundo
- Una banda ancha 1000 veces más rápida por unidad de área

- Hasta 100 dispositivos más conectados por unidad de área (en comparación con las redes 4G LTE)
- Disponibilidad del 99.999%
- Cobertura del 100%
- Reducción del 90% en el consumo de energía de la red
- Hasta 10 diez años de duración de la batería en los dispositivos IoT (Internet de las Cosas) de baja potencia.

5.5.5 Comparación entre los sistemas infrarrojos, FSO y Ultrasonidos.

Las señales infrarrojas poseen un gran ancho de banda de hasta 16 Mbps y por sus características también tienen fuerte resistencia a las interferencias, presenta como sistema un bajo consumo de potencia y permite el uso de componentes muy económicos; la desventaja se presenta ya que al no poder atravesar objetos, cualquier obstáculo impide las comunicaciones y también presentan alta sensibilidad a interferencias externas y a las condiciones atmosféricas que podrían incluso llegar a atenuar la señal. El alcance que presenta es otra desventaja pues es máximo de 5 metros y necesitan de línea de vista para comunicarse.

Los sistemas FSO tienen como principal ventaja la instalación sencilla que presentan, y que su infraestructura es relativamente barata, pero también necesita línea de vista entre los transmisores y receptores, al igual que el caso anterior, a pesar de no poder atravesar objetos, tienen una alta reacción a los cristales, por lo que se pueden colocar repetidores en caso de encontrar obstáculos representativos, son muy robustos en cuanto a la seguridad, ya que las emisiones laser no se pueden detectar mediante analizadores de espectros o medidores de potencia; lamentablemente presentan una alta vulnerabilidad hacia las condiciones atmosféricas, por otro lado es necesaria una perfecta alineación entre transmisor y receptor.

Finalmente, los sistemas por ultrasonidos tienen la ventaja de detectar objetos u obstáculos independientemente del color que sean, así como el consumo de potencia es bastante bajo, y se poseen sensores inmunes a la suciedad y el polvo. Fundamentalmente su desventaja está en que se deben usar un banco de sensores ultrasónicos para abarcar el perímetro de un robot.

5.5.6 Comparación entre Bluetooth, WiFi y ZigBee.

El sistema Bluetooth a pesar de ser el entorno adecuado para trabajar con robots móviles, pues, aunque su consumo sea mayor que los sistemas analizados en el anterior inciso, sigue siendo muy bajo (40 mA), además permite la comunicación bidireccional; posee la posibilidad de comunicación simultánea entre nodos proporcionando la ventaja de la multidifusión de la información, que estrategias colaborativas así lo requiere; está perfectamente establecido y disponible en una amplia variedad de dispositivos electrónicos, por lo que eventualmente se pueden usar estos de diferentes formas de repetición o transmisión de señales.

Su principal desventaja es la velocidad de operación, comparadas con otras tecnologías de corto alcance, además de que la seguridad no es tan buena como en los demás, que admite un código pin corto, que hace que se puedan producir efectos de suplantación de identidad.

WiFi en cambio presenta, posiblemente la más alta velocidad de transmisión de la información desde 11 Mbps, a sistemas de 1.3 Gbps, además de que, al igual que bluetooth permite comunicaciones bidireccionales, presentando como ventaja también su amplia difusión y penetración en el mercado de dispositivos electrónicos. Su principal desventaja frente a los otros sistemas es su consumo elevado de potencia y que los costos económicos de sus módulos son también más caros que sus contrapartes bluetooth y ZigBee.

Los sistemas ZigBee presentan bajos consumos de energía por su característica de permanecer en reposo, siendo mucho menor que los dos sistemas anteriormente mencionados, se incorpora además una ventaja adicional de la seguridad, pues emplea sistemas de encriptación para evitar intromisiones en la red; los módulos son de bajo costo económico en dependencia de su funcionalidad y existen en múltiples alternativas y variedades hasta de frecuencia de operación; y fundamentalmente, las redes de estos dispositivos pueden aumentarse coexistiendo en ellas hasta 65000 nodos que es la mayor ventaja en este aspecto por sobre todos los demás sistemas.

Como desventaja tiene una velocidad de transmisión relativamente baja y la cobertura es de menor distancia, pero, para aplicaciones de robótica móvil, en las cuales las distancias de operación en colaboración son pequeñas, es un sistema ideal.

El resumen técnico entre las tres tecnologías se presenta en la Tabla 5.3.

Tabla 5.3. Comparación entre tecnologías inalámbricas de red.

Parámetro	Bluetooth	WiFi	ZigBee
Velocidad	1 Mbps	11 Mbps	250 Kbps
Alcance	10m	100m	90m
Consumo	40 mA	250 mA	30 mA
Tipo de Red		WLAN	WPAN

5.5.7 Comparación entre sistemas GSM; GPRS y 3, 4, 5G

Como se había mencionado GSM realiza comunicaciones con el sistema de conmutación de circuitos presenta un retardo menos que el de conmutación de paquetes, presenta alta eficiencia y el ancho de banda que se usa para la comunicación siempre está disponible, que se considera como desventaja cuando no se requiere comunicaciones, pues ese ancho de banda está reservado, eso implica que el retardo es menor se compensa de forma negativa con el tiempo de establecimiento y cierre de la conexión.

Los sistemas GPRS aprovechan mejor el ancho de banda, y además permiten cambiar de velocidad de transmisión y escritura de los datos, la conmutación de paquetes permite manejar adecuadamente el tráfico y congestión de los caminos de comunicaciones, pero introducen retardos mayores por las colas de espera en cada nodo.

Finalmente, la clara ventaja de los sistemas XG, es su enorme velocidad de transmisión y alcance, pudiendo llegar a cientos de kilómetros, pero en desventaja también hay sitios en donde no se posee de forma definitiva esos enlaces “nuevos” de comunicaciones.

En conclusión, de todo lo escrito, la Tabla 5.4 muestra en resumen las características técnicas que se han tomado en cuenta para la elección del sistema inalámbrico de comunicaciones en las aplicaciones con robots móviles.

Tabla 5.4. Comparación entre redes consideradas para las aplicaciones en robótica móvil

Parámetro	Instalación	Alcance	Velocidad Transmisión	Consumo Transmisión	Consumo Reposo	Costo
IR	Baja	1-5 m	2.4 Kbps – 16 Mbps	30 mA	9 mA	Bajo
FSO	Alta	50-500 m	10 Mbps – 1 Gbps	30 mA	-	Alto
US	Alta	1-10 m	160 Mbps	35 mA	3 mA	Medio
Bluetooth	Media	1-15 m	9.6 Kbps – 23 Mbps	40 mA	0.2 mA	Bajo
WiFi	Media-Alta	20-100 m	11-100 Mbps	250 mA	13 mA	Medio-Alto
ZigBee	Baja	1-100 m	250 Kbps	30 mA	3 μ A	Medio
GSM/GPRS	Alta	-	9.6-114 Kbps	200 mA	1.5 mA	Medio-Alto
XG	Alta	-	380 Kbps - 1 Gbps	1500 mA	-	Alto

Para el escenario de operación: en primer lugar, y por todo lo que se había explicado en el Capítulo III, es necesaria la comunicación entre la estación principal de control, que dependiendo de las aplicaciones a las que será sometido el robot y toda esta plataforma, contendrá los algoritmos de control de movimiento del robot, y los algoritmos de la aplicación particular: búsqueda, localización, detección, colaboración, etc. Y por otro lado el robot en, sí poseerá una plataforma de control de estabilidad y comunicaciones, que le harán responder a las órdenes del sistema según lo que se necesite y en tiempo real, que viene a ser la principal restricción del sistema de comunicaciones a implementar.

Esa es la primera aproximación; la segunda y no menos importante es la presencia de más robots para realizar el trabajo colaborativo, ésta también dependiente de las aplicaciones colaborativas, pero en general se necesita de una red de comunicaciones entre los robots y la estación central, que nuevamente contiene la inteligencia necesaria para su aplicación, y la comunicación entre robots, añadiéndole la necesidad de un controlador de la red.

Por lo dicho y por las principales características hechas en la comparativa de la Tabla 5.4, la elección del sistema a utilizar es la red ZIGBEE, pues es rápida, tiene un alcance suficiente para tareas colaborativas, la adquisición de los módulos es conveniente, su instalación y configuración muy intuitiva y además es móvil por completo y adiciona un consumo de energía casi tendiente a cero cuando su estación no está activa, que es importante en los algoritmos de colaboración y finalmente está ampliamente difundida es barata y sus módulos se encuentran en el mercado global completamente disponibles.

A continuación, se destaca la forma y configuración posibles (puede ser diferente para cada aplicación) de las redes a implementarse en los robots con ZigBee.

5.6 Desarrollo de la red ZigBee en los robots contruidos.

Como se manifestó anteriormente, los módulos XBee que permiten implementar una red ZigBee presentan un menú completo de capacidades de comunicaciones, desde los módulos básicos de la serie 1 hasta los módulos avanzados de series 2 y 3, formando un verdadero ecosistema (así lo llaman los fabricantes) de módulos y sistemas para comunicaciones y en bandas de frecuencia de entre los 2.4 GHz hasta, pasando por bandas LTE 2, 4 12, 13; LTE-M, hasta llegar actualmente a productos IoT.

En este proyecto se usan módulos de bajas prestaciones y de bajo costo, pues los requerimientos no son muy críticos y los ambientes en los que trabajan los robots (ambientes de prueba y laboratorio) son de bajas interferencias y degradaciones de las señales.

Las características de los módulos usados de forma general son:

- Alcance máximo: 100m.
- Potencia de transmisión: 1mW.
- Sensibilidad del receptor: -92 dBm.
- Consumo transmisión: 45 mA.
- Consumo recepción: 50 mA.
- Consumo modo inactivo: menos de 10 μ A.
- Uso de comandos AT para configuración de los módulos.
- Amplio set de comandos de configuración.
- Software de configuración gratuito.
- Aplica cerca de 65000 direcciones de red
- Soporta comunicaciones punto a punto, punto a multipunto y broadcast.
- Pequeño tamaño.
- Gran portabilidad.
- Bajos voltajes de suministro.

Para la conexión eléctrica de estos módulos generalmente se usan únicamente las señales de alimentación, transmisión y recepción. En las placas de los robots construidas, se consideró la conversión de la fuente regulada que pueda entregar hasta 12 Vdc, las señales de alimentación de los módulos XBee de 3.3 Vdc.

5.6.1 Tipos de nodos a configurar:

De alguna forma mínima ya se habló en las características de la red de los tipos de nodos que se pueden configurar: coordinadores, enrutadores y nodos terminales; El nodo **coordinador** tiene la función fundamental de establecer la red y controlarla, establece el canal de funcionamiento y el PAN ID para la red, después de lo cual los demás dispositivos podrán adjuntarse a ella, ya sean routers o dispositivos finales y asociarse a esa red.

Los **Router** crean y mantienen las comunicaciones y direccionan las rutas para la transmisión de los paquetes de comunicaciones entre nodos, Un router debe unirse a una red, antes de permitir otros dispositivos se unan a él; ya sean otros routers o dispositivos finales.

Los **dispositivos finales**, o terminales no tienen capacidad de enrutamiento, sino únicamente de transmisión y recepción de información dentro de la red; siempre deben interactuar con un nodo principal.

a. Protocolo ZigBee.

Usa redes de tipo PAN, incluye un coordinador y uno o más routers y uno o más dispositivos finales, se configura de acuerdo a parámetros de máximo ancho de la red, número de routers y número de dispositivos finales máximo.

El proceso de asociación es el cual en el que, un nodo se une a una red ZigBee, de tipo PAN, siendo parte de ella como coordinador o router; esto lo realiza asignando la dirección de 16 bits del nodo que se asocia; el ancho máximo de la red es de 5, siendo el máximo número de routers 6 para una PAN MaxStream, al igual que se pueden integrar a esta forma un máximo de nodos finales de 14; La dirección de red es una palabra de 16 bits que se asigna a un nodo después de que se ha asociado a un coordinador, y el coordinador encendido será el que determine el canal de operación, y se establecerá una tabla de enrutamiento.

Una vez que el router se ha incorporado a la red, también puede permitir que otros nodos se asocien a él. La asociación establece una relación padre/hijo entre dos nodos. El nodo que permite la unión es el padre y el nodo que se une es el hijo. La relación padre/hijo no es necesaria para el enrutamiento; sin embargo, es necesario para la formación de la red y la asignación de direcciones de red. Si un coordinador no existe, la red no puede ser formada. Un nodo no puede transmitir o recibir datos hasta que se una a un PAN.

- b. Funciones y operación del coordinador ZigBee:** Para formar una red, el coordinador selecciona un canal de operación que no esté en uso, e identificar el nombre de su red conocido como PAN_ID; para esto, se realiza una exploración de la energía de todos los canales; el tiempo de exploración de cada canal está establecido por un parámetro llamada SD (Scan Duration), cuando se ha terminado este proceso una bandera llamada Active Scan se produce, devolviendo una lista de coordinadores y routers encontrados.

Si el parámetro ID (PAN ID) = 0, el Coordinador seleccionará un ID aleatorio para la PAN. De lo contrario, el Coordinador almacenada su parámetro de identificación.

Después de que el Coordinador se ha iniciado, permitirá que los nodos se asocien a él por un tiempo en función de su parámetro NJ (Node Join Time- Tiempo de Asociación Nodo). El canal de operación y PAN ID se pueden leer con los comandos CH (canal de operación) y ID (ID PAN). La dirección de 16 bits del Coordinador es siempre 0x0000.

Si un coordinador ha formado una red, se mantendrá la información del PAN ID y del canal de operación, así como una lista de sus nodos hijos. Sin embargo, si cualquiera de los parámetros SC (Escanear Canales) o ID (ID PAN) han cambiado y el Coordinador se reinicia, el Coordinador inicializa con el nuevo valor de SC y/o ID y se borrará la lista de nodos hijos.

- c. Direccionamiento en redes ZigBee:** La dirección de red de 16-bit se asigna a un nodo cuando se asocia a una red. La dirección de red es única para cada nodo en la red. Sin embargo, las direcciones de red no son estáticas, pueden cambiar.

Las siguientes dos condiciones pueden hacer que un nodo reciba una nueva dirección de red:

1. El dispositivo final no puede comunicarse con su padre.
2. Un router o dispositivo final, envía un comando de notificación de que está huérfano para localizar a sus padres. Si el nodo principal no responde, el router o dispositivo final considera que no está asociado y repite el proceso de unión a la red. Una vez que el nodo se une a la red, este y todos sus descendientes recibirán una nueva dirección de red.

Dado que todas las comunicaciones ZigBee usan direccionamiento de 16-bit, la dirección de 16 bit del nodo debe ser conocida antes de que las comunicaciones pueden tener lugar.

- d. Direccionamiento de los módulos XBee:** Cada módulo de RF tiene una dirección única de 64 bits asignada de fábrica que se puede leer usando los comandos SH (Número de serie High) y SL (Número de serie Low). Además, cada módulo puede almacenar una cadena de identificación usando el comando NI (Node Identifier-Identificador del Nodo).

Los módulos RF XBee pueden direccionarse utilizando su dirección de 64 bits (de fabrica), su cadena de identificación NI o ambas direcciones de red de 64 bits y de 16 bits. En el protocolo ZigBee, la dirección de red de 16-bits de un coordinador es siempre "0". Los nodos pueden direccionarse al Coordinador utilizando esta dirección.

Para enviar un paquete a un módulo RF utilizando su dirección de 64 bits (modo transparente): establezca los parámetros DH (Dirección de destino High) and DL (Dirección de destino Low) del nodo origen para que coincida con la dirección de 64 bits (parámetros SH (Número de Serie High) y SL (Número de Serie Low)) del nodo de destino.

Dado que el protocolo ZigBee se basa en direcciones de red de 16 bits para el enrutamiento, la dirección de 64 bits debe convertirse en una dirección de red de 16 bits antes de la transmisión de datos. Si un módulo no sabe la dirección de red de 16 bits de una dirección dada de 64 bits, se transmitirá un broadcast mediante el comando Network Address Discovery (descubrir direcciones de red). El módulo con una correspondencia con la dirección de 64 bits transmitirá nuevamente su dirección de red de 16-bit. Los módulos mantienen una tabla que puede almacenar hasta siete direcciones de 64 bits y sus correspondientes direcciones de red de 16-bits.

Por ser un tema amplio, las características más específicas de las comunicaciones configuradas, se especificarán en cada aplicación.

5.7 Diseño de la plataforma de supervisión.

Tal y como se había mencionado anteriormente, al finalizar el trabajo de fin de máster, en este trabajo continuo con la robótica móvil, se había planteado la necesidad de: además de realizar todo lo que ya se ha expuesto en esta tesis hasta el momento, también implementar un sistema de software que permita realizar aspectos de supervisión y teleoperación de los robots involucrados, especialmente cuando ellos ya están fuera de recuperación, es decir perdidos por completo en la operación y consecución de sus tareas.

Para ello originalmente se había desarrollado una plataforma inicial, en el software LabVIEW de National Instruments; esta elección se desarrolló, pues las plataformas conectadas al robot operaban en ambiente Debian, y se controlaban de forma automática, pero su programación, se la realizaba desde un computador con sistema operativo Ubuntu y distribución Debian también, a partir de las conexiones SSH y de hecho las pruebas de los programas ejecutables también se las realizaba desde allí; los programas desarrollados y el algoritmo de movimiento y el de control del robot, fueron realizados en lenguaje C y ejecutados como antes se mencionó.

Al ser LabVIEW un software que permite fácilmente desarrollar interfases gráficas de usuario y HMI's, además de tener un set enorme de recursos de programación, y principalmente que su

plataforma se ejecuta en Windows, IOS y Linux, se vio la posibilidad de realizar los algoritmos en este lenguaje y crear un programa ejecutable en la distribución Debian. Lamentablemente surgieron dos inconvenientes : en primer lugar, las distribuciones sobre las que se ejecutaba LabVIEW con toda su potencia y versatilidad eran únicamente Red Hat, Fedora y Mint, más no Debian ni Ubuntu; en segundo lugar, el lenguaje de programación gráfico de LabVIEW y el uso de toolkits referentes no se desempeñaban como se quería, es así que para el desarrollo de algoritmos la programación no era muy flexible; a pesar de que las versiones actuales tienen un utilitario que abre la posibilidad de escribir código de MATLAB en él y ejecutarlo con lo demás en ambiente LabVIEW.

Estas opciones se comprobaron, pero se introducían retardos considerables al momento de ejecutar los programas; entonces la solución fue usar MATLAB en las computadoras centrales desarrollando y ejecutando los algoritmos de control y movimiento y comunicarse con LabVIEW para que desarrolle solo la interfase, en virtud de que en MATLAB la creación de interfases es un poco complejo y además no tiene todas las herramientas adecuadas para hacerlo.

El procedimiento mencionado fue adoptado inicialmente pues la aparición del módulo de robótica en LabVIEW permitió contar con más herramientas de desarrollo de las aplicaciones y mejor comprensión de las interfases, que se pueden desarrollar a través de este módulo de forma más intuitiva. En la Figura 5.2 se puede observar una interfase aplicando esta tecnología.

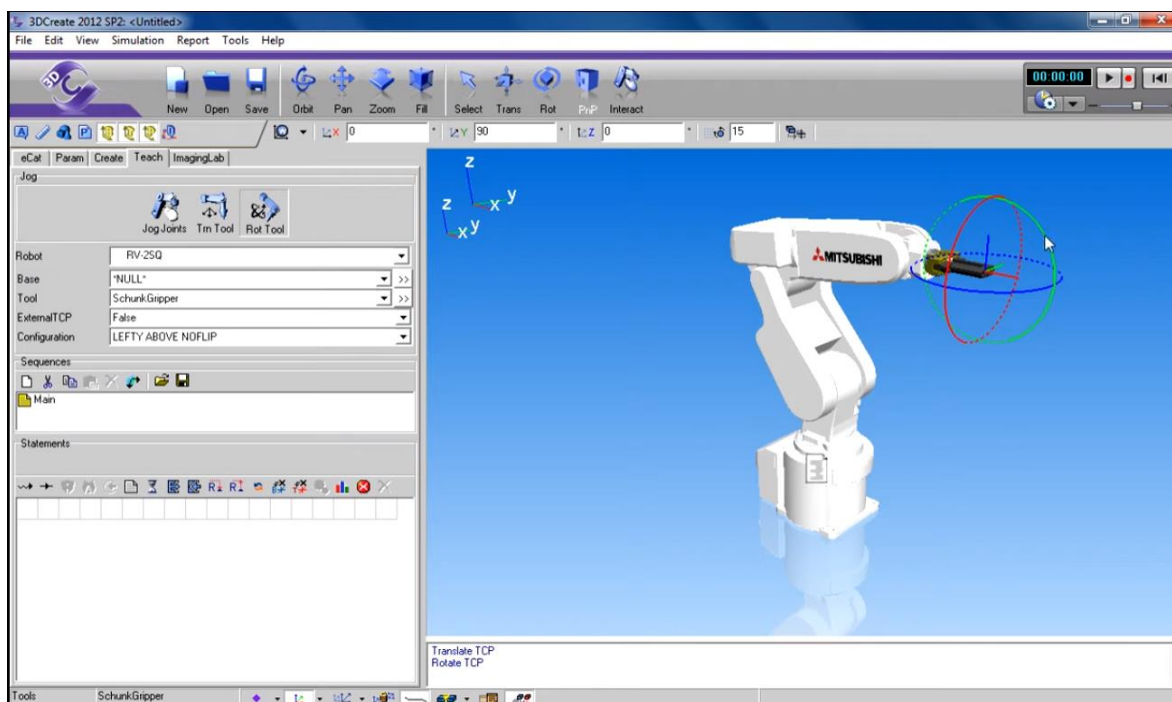


Figura 5.2 Entorno de trabajo en LabVIEW Robotics.

Como se puede observar a pesar de que la interfase puede realizarse en tres dimensiones y obtener una penetración considerable, los programas son hechos en MATLAB y estos interactúan al mismo tiempo que con el robot y con LabVIEW y se generan retardos innecesarios, además de que no se puede navegar por el ambiente de aplicación que será supervisado, es un ambiente en 3D que resulta muy estático.

Entonces se investiga realizar las interfases en un lenguaje de programación que desarrolle soluciones de realidad virtual y desarrollar las interfases directamente con él y arrastra los datos en tiempo real, vía DDE, COM, TCP, etc. De esta forma se encuentra Unity y se desarrolla el sistema de supervisión en él.

5.8 Creación de la interfase de supervisión 3D.

La integración de los dos entornos computacionales exige la integración de medios y dispositivos, por un lado, el (o los) robots autónomos, por otro lado, la computadora de control y corriendo en forma paralela el sistema de realidad virtual. Es necesario empaquetar la información proveniente de las fuentes mencionadas y embeberle en una aplicación de desarrollo, programada y ejecutada en Unity.

El punto neurálgico del desarrollo es de forma definitiva el motor gráfico de Unity, pues es el entorno alrededor del cual suceden los eventos virtuales, se usa una versión 5.3.5 para mejorar la compatibilidad con todos los elementos del entorno.

Para desarrollar la interfase de usuario que se extenderá a convertirse en una HMI, se debe crear y animar los objetos 3D y crear los entornos virtuales en donde se le va a colocar.

5.8.1 Importación de un modelo 3D.

Estos modelos generalmente se pueden obtener de proyectos realizados en software de CAD, en SolidWorks generalmente, y es una ventaja porque los modelos de los robots y sus simulaciones se realizaron también en el mismo software. En este apartado hay que destacar que el modelo generado se lo debe grabar con formato de archivo de ensamblaje (.asm). En la Figura 5.3 se observa el modelo del manipulador móvil diseñado en SolidWorks y que se va a importar en Unity, para su animación y control.

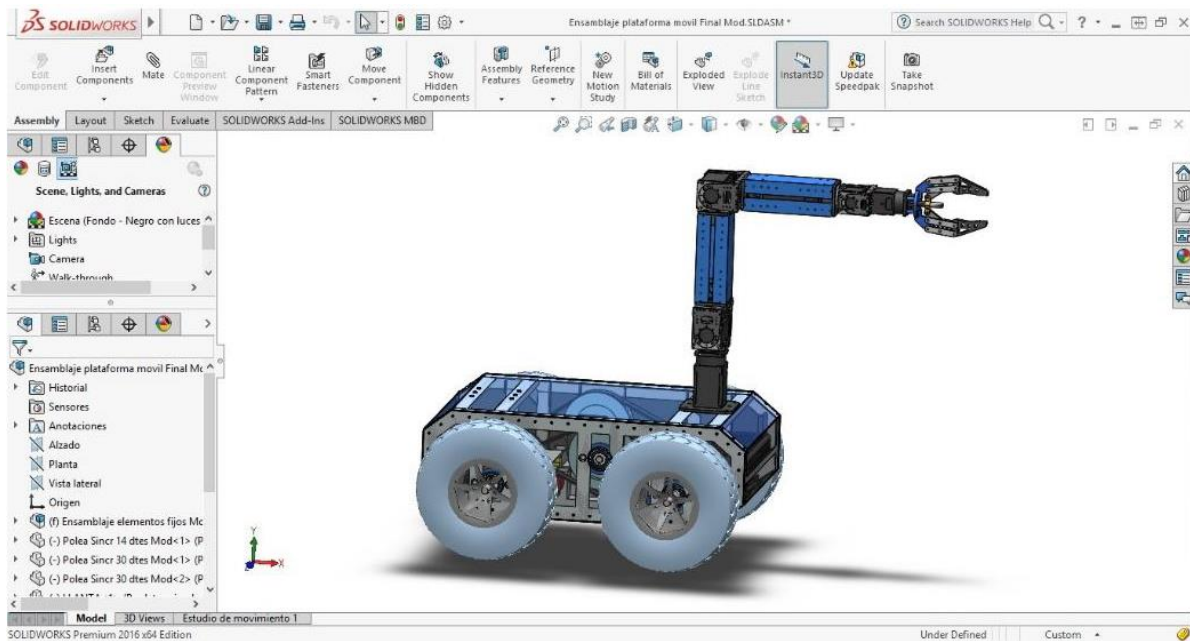


Figura 5.3 Modelo del robot manipulador disponible para importarlo a Unity.

Como paso obligatorio se le exporta a este modelo, con todas sus características (dimensiones y atributos) al programa 3ds Max, para establecer jerarquías de cada elemento insertado para formar el objeto 3D. La organización de sus componentes en forma de jerarquías, asegura su perfecta compatibilidad en la simulación del movimiento, ya que se superponen en un mismo espacio cada una de estas formas en jerarquía, esto se observa en la Figura 5.4.

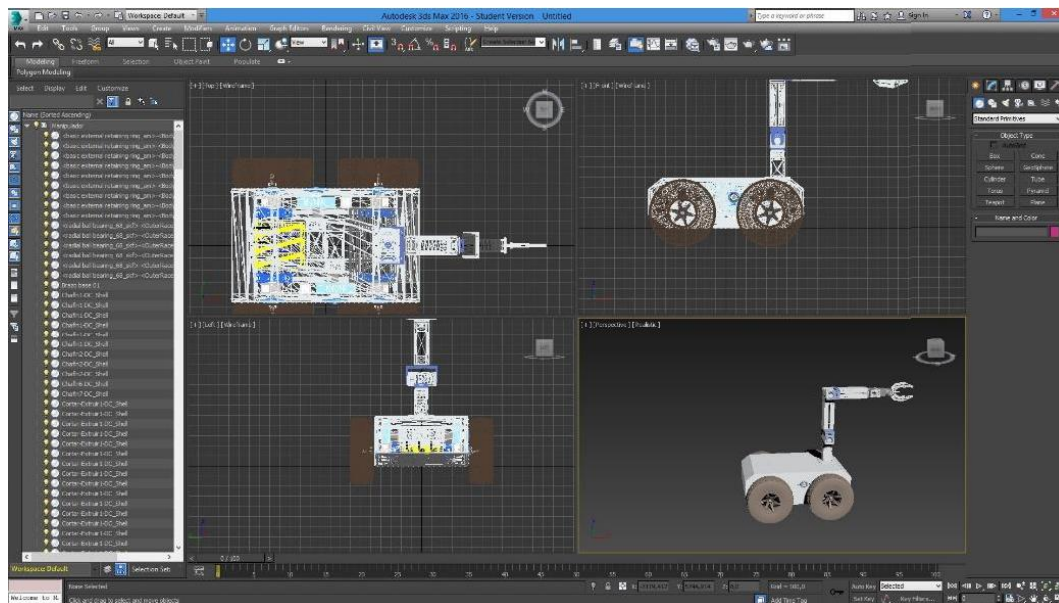


Figura 5.4. Modelo 3D del manipulador móvil, insertado por jerarquías en 3DMax

El resultado de este proceso deriva en la importación del mismo en formato (.fbx) y de allí poder importarlo finalmente a Unity, como se observa en la Figura 5.5; en esta instancia se le agregarán ya elementos especiales como texturas especiales o entornos digitalizados, según las capacidades implementadas en Unity.

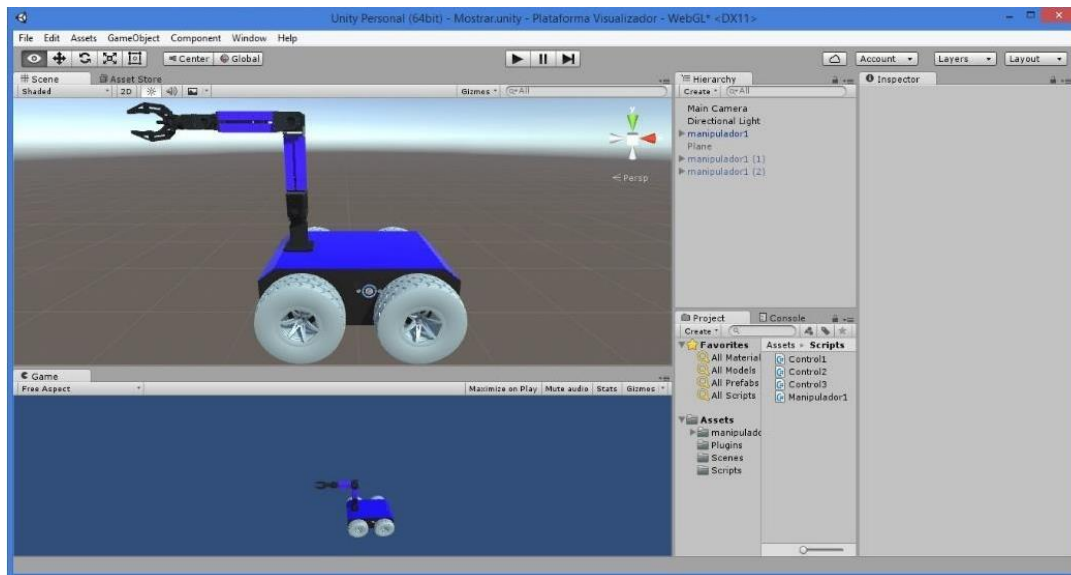


Figura 5.5 Modelo del manipulador móvil insertado en Unity.

Para darle el movimiento al robot dentro de la aplicación en 3 dimensiones, se usa el procedimiento real, que es el uso del modelo cinemático, especificando los movimientos de rotación y los movimientos de rigging, que es el proceso de crear un sistema de controles digitales y agregárselos a un modelo 3D para que así pueda ser animado fácilmente y de forma eficiente.

5.8.2 Movimiento de rotación.

Se caracteriza por mover independientemente cada articulación de un modelo 3D, despreciando los demás movimientos de los demás elementos, aunque la lógica que controla el conjunto global del modelo debe tomar en consideración todos los puntos de rotación para integrar y logra el objetivo de visualización del modelo 3D.

En la Figura 5.6 podemos observar el modelo 3D de un robot manipulador, componente superior del manipulador móvil, al cual se otorgarán movimientos por punto de rotación; la configuración del script de lógica de ejecución debe tener en cuenta será única para este movimiento de cada articulación del brazo, este modelo debe pasar por todas las articulaciones hasta encontrar la deseada y aplicarle el movimiento.

Este movimiento no se controla desde Unity, pues en este software solamente se lo simula y representa, es el MATLAB quien tiene un algoritmo interno que controla este movimiento, que

como de ha dicho anteriormente depende de su cinemática interna ya desarrollada. En la Figura 5.6 se observa este apartado.

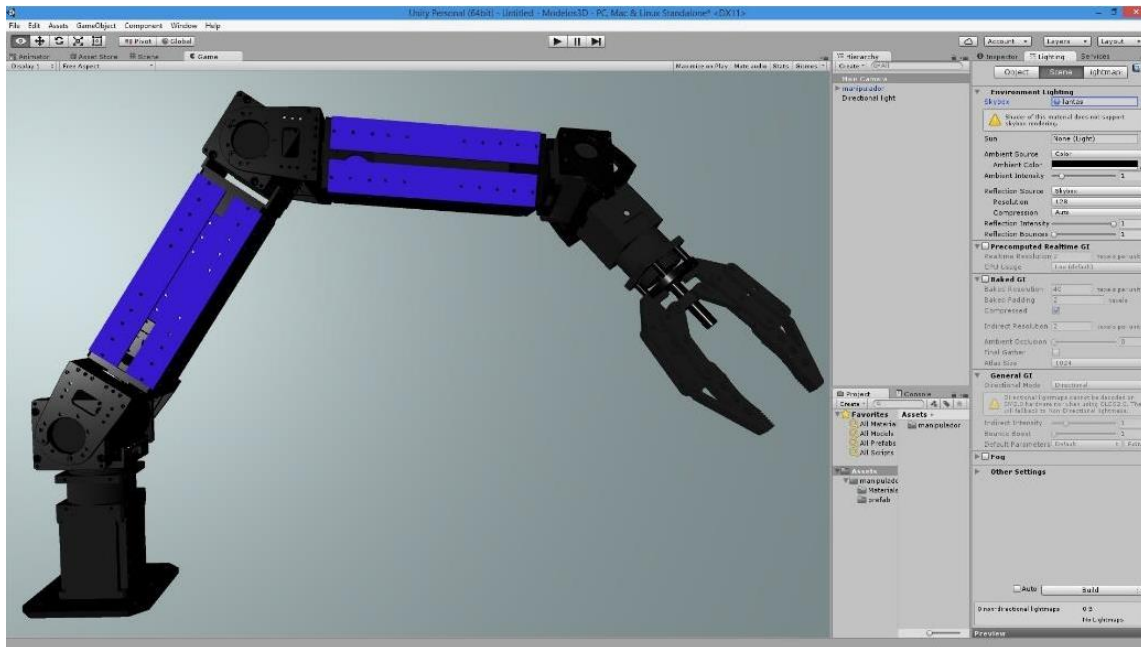


Figura 5.6. Lógica de operación del robot manipulador en Unity.

5.8.3 *Movimiento de Rigging.*

Este se usa para conseguir se emule de la mejor forma el modelo cinemático de los robots, este procedimiento no es más que la transmisión del movimiento del eje de un motor hacia las ruedas de tracción y su respuesta al ambiente en el que giran, en el robot manipulador estos motores giran hasta llegar al extremo operativo según la tarea a desarrollar; así como por ejemplo: un robot de modelo Theo Jansen, en el que, un solo motor hace girar o trabajar un sinnúmero de patas articuladas que mueven el conjunto total a un objetivo definido.

Para poder proporcionarle el movimiento adecuado es necesario la interiorización del modelo de movimiento que usas el modelo del robot, entonces, se dibujan los elementos fundamentales en cada instancia de movimiento desde el motor al extremo móvil que depende en este caso si es el robot unicycle, el omnidireccional o el manipulador móvil.

El modelo 3D en Unity, lo que hace es crear una secuencia de estos dibujos para producir una animación, solapando en el tiempo, cada vez una imagen; estas animaciones dependen mucho de la cinemática del movimiento del robot, ya que muchas veces según ella serán necesarias imágenes desarrolladas cada grado sexagesimal de rotación o unas pocas de ser el caso; así mismo, si no se necesita esa “precisión” de movimientos, se realizara una captura de las

imágenes o dibujos cada cierta posición relativa; p.e. Un movimiento horizontal desde una vista x, solamente requiere de una imagen desplazada en el entorno del eje x.

5.8.4 Ambiente Unity en el objeto de simulación.

Al ser Unity una herramienta diseñada para desarrollar juegos en realidad virtual y ambiente 3D, permite la creación de entornos de campo a partir de objetos prediseñados que están a disponibilidad completa del programador que adquiera la licencia. El motor Unity ofrece una amplia gama de características y posee una interfaz sencilla de entender. Además, posee un sistema de integración multiplataforma que permite exportar juegos para casi todas las existentes, siendo una de las mejores opciones para desarrollo 3D en plataformas de Android y otros sistemas operativos; por sus herramientas de compresión que permiten que los videojuegos no sean especialmente pesados, y no consuman excesivos recursos.

Unity para el desarrollo de los videojuegos tiene, materiales, texturas, elementos, que pueden proporcionar (y ese es el objetivo), más realismo a las aplicaciones, para ello es necesario la importación de ambientes de desarrollo (Assets>>Import package>>Enviroment), este proceso lo observamos en la Figura 5.7. a y b.

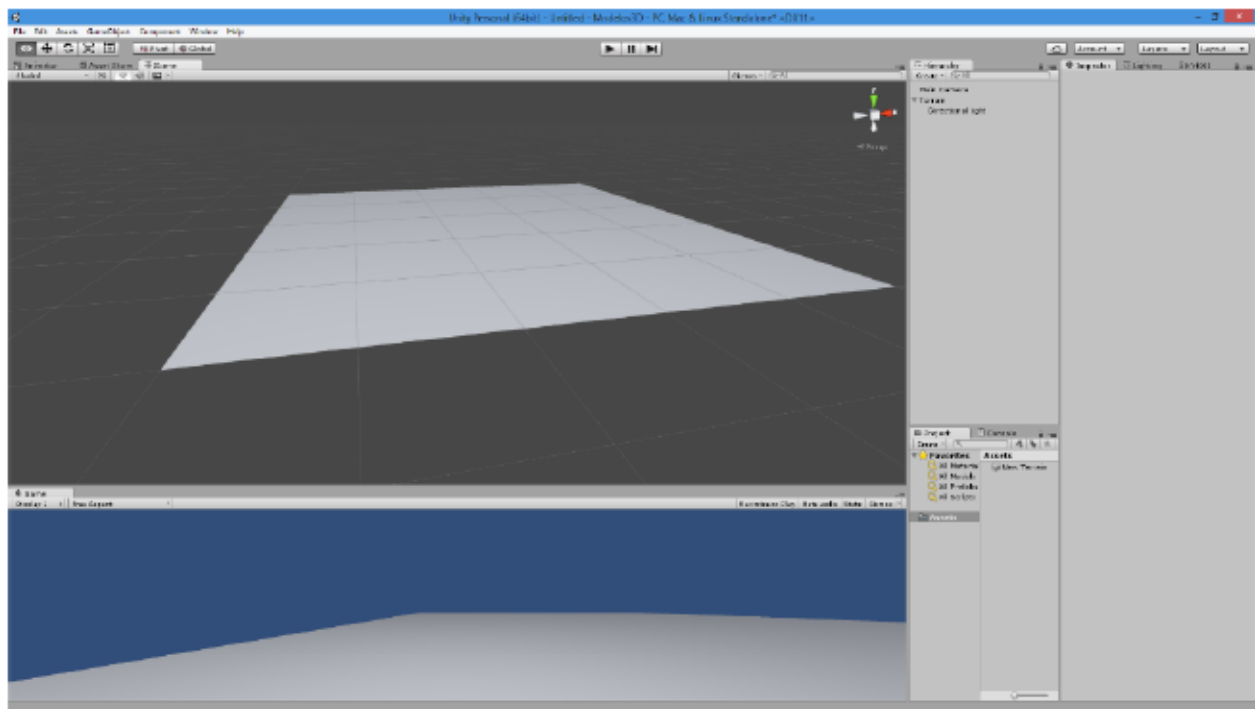


Figura 5.7. (a) Entornos virtuales en Unity.

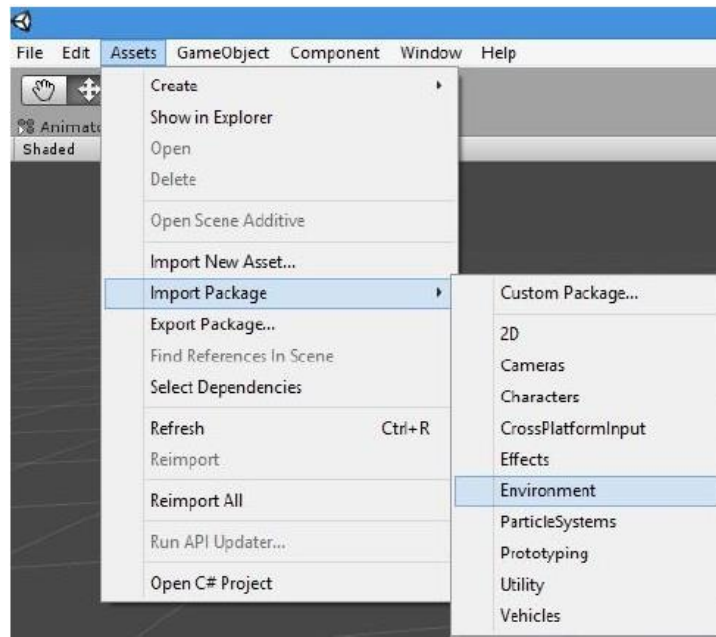


Figura 5.7. (b) Obtención de ambientes variados en Unity

Si se desea agregar, a partir de los elementos básicos otros objetos como: deformaciones, arboles, suelos diversos, etc., se seleccionan desde la carpeta inspector, como lo ilustra la Figura 5.8.

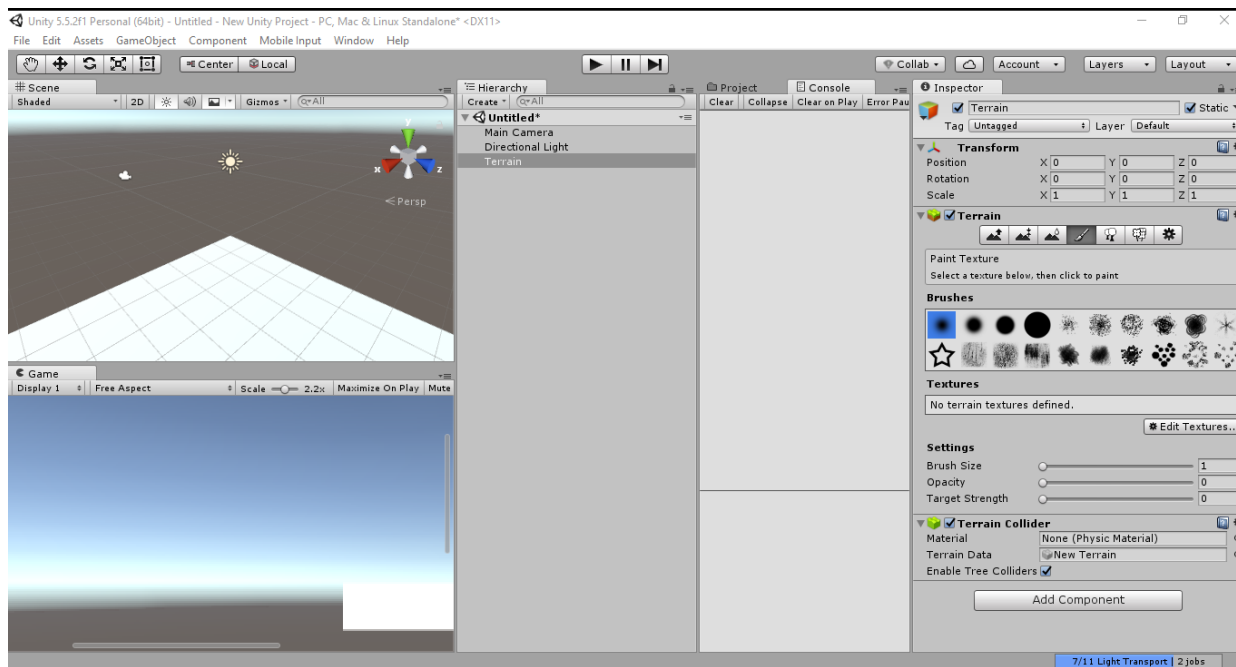


Figura 5.8. Selección de herramientas de dibujo en Unity.

Componiendo una combinación de elementos de entre árboles, montañas, depresiones, lagos, etc., que se pueden incluir, el resultado del ambiente 3D de Unity puede observarse como un terreno virtual completo como en la Figura 5.9.

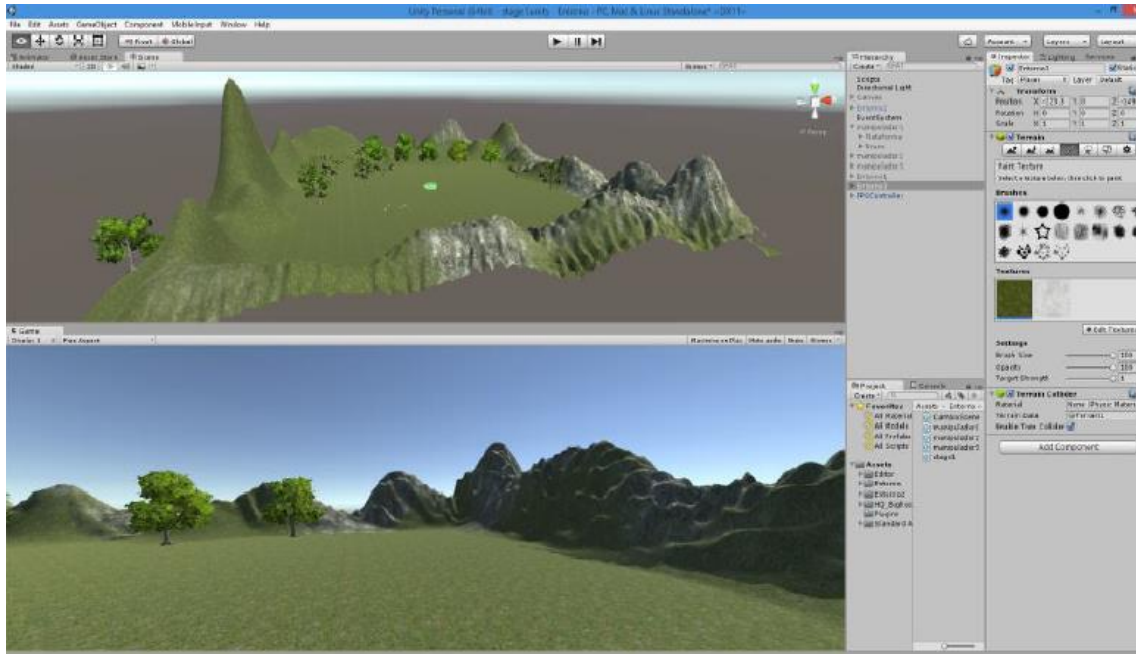


Figura 5.9. Ambiente virtual en 3D construido a partir de Unity.

5.8.5 Interacción del ambiente con las aplicaciones.

A partir del ambiente construido y simulado se pueden elaborar tres procedimientos importantes, que son inclusive componente filosófico de las HMI (**H**uman **M**achine **I**nterface); y que son los procedimientos de monitoreo, control (o en este caso la operación remota) y la supervisión general del sistema; estos conceptos extraídos de la automatización de procesos industriales.

Para el monitoreo, lo único que hace falta es interconectar en tiempo real, todos los datos de entrada a la interfase Unity y observarla de forma intuitiva y en 3D; es decir no se podrá, en este caso, la intervención en el proceso de forma manual, sino únicamente la observación de los que esté ocurriendo en el entorno, vuelvo a repetir, con la información adecuada necesaria producida en ese momento mismo de la conexión con la aplicación 3D, en este caso para las tres aplicaciones, la conexión de los datos se hace entre Unity y MATLAB, ya que los algoritmos de control de las aplicaciones y el movimiento provienen del PC central en donde corre también la aplicación Unity, esta conexión e intercambio de datos puede hacerse a través de una conexión COM o DDE creando las DLL adecuadas como primera opción; la interacción con un servidor de OPC que administre las variables, o la conexión vía TCP/IP entre las dos aplicaciones y de forma transparente.

Se han usado las tres opciones en todas las aplicaciones, y como se ha destacado muchas veces en este trabajo, la opción más conveniente, está en dependencia de la aplicación que se desarrolle, pues en los casos más críticos, la conexión e intercambio de la información no debe sobrepasar de un determinado intervalo de tiempo (100mS), para poder emplear adecuadamente los algoritmos de aplicación, si no, los movimiento de los robots son completamente erróneos y se incrementa el error en la aplicación en porcentajes verdaderamente altos; en los demás casos, (cuando solo se incorpora un solo robot generalmente), los retardos producidos por este sistema es despreciable.

En forma general se puede decir que para todas las aplicaciones desarrolladas, este sistema simula perfectamente, lo que está pasando en el ambiente de aplicación de los robots móviles; para ello la interfaz de usuario (GUI, que se puede transformar, según lo dicho anteriormente en una HMI) se construye a partir del apartado User Interface de la escena principal de Unity; cuando se quiere realizar la simulación o supervisión (como elementos mínimos) del sistema de aplicación, la única conexión a realizar es la de Unity y MATLAB; cuando se desea realizar la operación remota, es decir el mando a distancia de los robots, desde la estación Unity, los datos deben considerarse de entrada y salida, es decir la acción iniciada en Unity deberá reflejarse en los robots, pasando de forma transparente por el algoritmo MATLAB (que controla el movimiento) y finalmente en la supervisión se emplearan funcionalidades automáticas para la realización de aplicaciones, con la anuencia del supervisor, es decir acción de operador en la HMI, hasta el límite de realizar un shutdown de todo el proceso y en tiempo real.

5.8.6 Conexión entre Unity y MATLAB (y otras aplicaciones).

Como se ha mencionado antes; el cerebro del sistema es un ordenador principal, en donde se ejecutan los algoritmos inteligentes; este código ha sido desarrollado en software MATLAB por su versatilidad matemática especialmente; esto quiere decir que las variables resultantes de estos procesamientos inteligentes y que son las que gobiernan los robots, se deben también reflejar en el software Unity; esta comunicación por defecto debe ser unidireccional (en monitoreo) pero a partir de la operación remota y más aun de la supervisión, los datos deben ser bidireccionales; es decir ambas aplicaciones gestionan los datos al mismo tiempo y por el mismo canal. Para ello es determinante el uso de memoria compartida; la memoria compartida (Shared Memory SM) se ejecuta por una librería de enlace dinámico de datos o DLL, que gestiona el espacio de memoria que se compartirá, definiendo permisos para las aplicaciones, etiqueta el espacio RAM de memoria y proporciona funciones para obtener información compartida y liberar el espacio libre cuando se ha cerrado la aplicación.

En la Figura 5.10, se observa la aplicación de creación de memorias compartidas; estas se pueden realizar también en una aplicación que corra con una base de lenguaje C++ y desarrollando un script, en este caso se usa una aplicación para su definición.

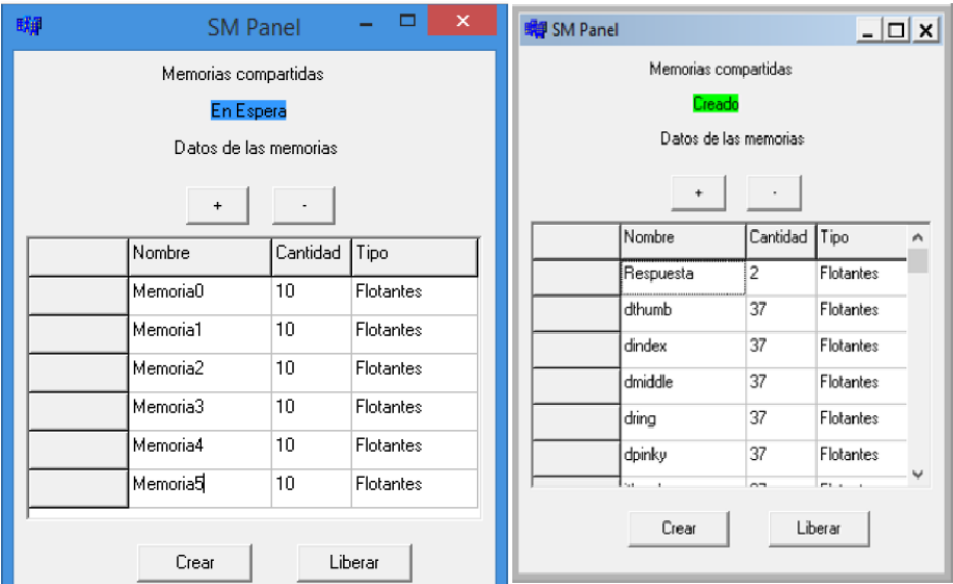


Figura 5.10. Interfase de definición de memorias compartidas entre aplicaciones.

Con estos espacios compartidos, lo demás es añadir códigos específicos tanto en MATLAB como en la aplicación de Unity para realizar la interconexión y el intercambio de datos. Lo fundamental es que se produzca una interacción de los componentes. La Figura 5.11 ilustra la conexión entre Unity y MATLAB.

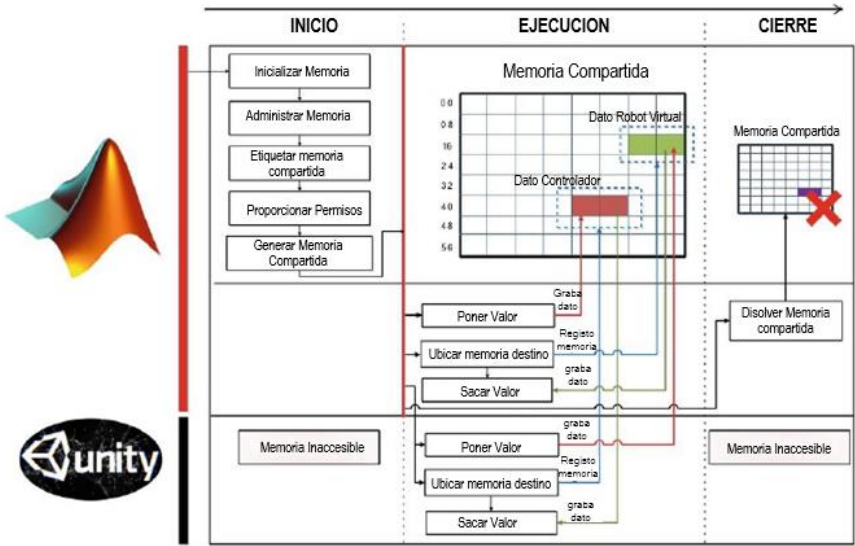


Figura 5.11. Intercambio de datos entre aplicaciones vía memoria compartida SM.

La operación remota del sistema, se realiza a través del intercambio de la información entre el sitio local y el robot o sistema que se encuentran a distancia. De acuerdo a lo señalado en la Figura 5.12.

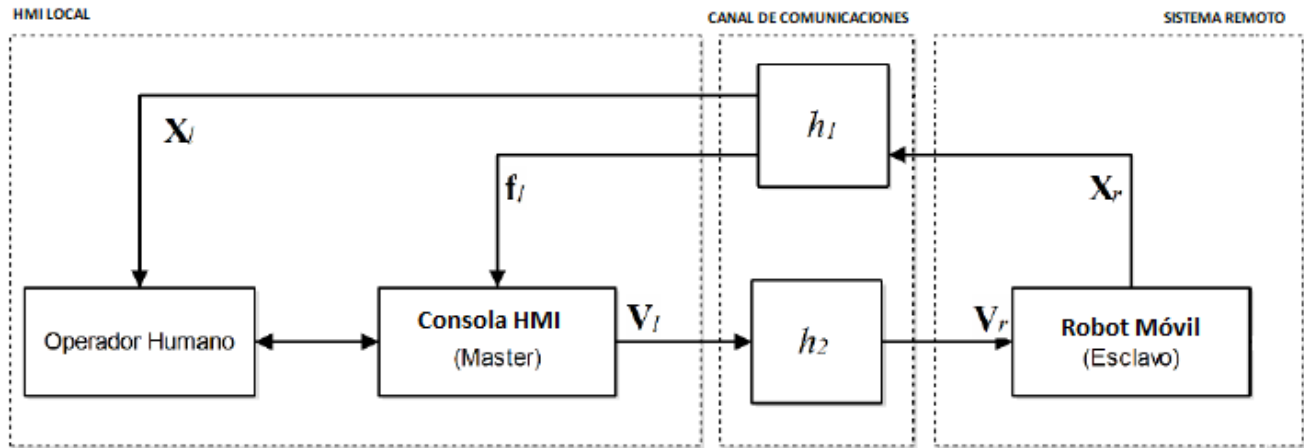


Figura 5.12 Diagrama de operación remota del sistema de Supervisión.

Este sistema consta de la HMI (por actuar sobre el elemento remoto en mando a distancia) sobre la que actúa el operador; en Unity se han proporcionado todos los elementos necesarios para su interacción, como sliders, botones, etc.; de acuerdo al movimiento que quiere realice el robot; el canal de comunicaciones sobre el cual se realiza la modulación, adaptación y transmisión de los paquetes entre el robot remoto y el ordenador donde reside su control, y el dispositivo operado a distancia que puede ser el robot móvil unicycle, omnidireccional o manipulador móvil. Este dispositivo actúa como el mando a distancia de un juguete, que hace que se dirija al sitio en que los controles le ordenen, pero en formato automático, el movimiento estará definido por la tarea que va a realizar; p.e: en un sistema de SLAM un robot ha trazado ya el mapa del ambiente y determina el objetivo a conseguir, y el algoritmo inteligente que ya ha procesado el mapa emplea un método de navegación (podría ser RRT) para conseguirlo, pero al ver el operador remoto la existencia de un obstáculo dinámico evita que el robot colisione y le “obliga” a hacer una trayectoria diferente de la predefinida, haciendo así valiosa la presencia del mando a distancia, de no ser así el algoritmo automático guiará al robot hacia su destino.

Capítulo VI

6 APLICACIONES, PRUEBAS Y RESULTADOS.

Como se ha podido leer en los capítulos anteriores, en este trabajo, se han diseñado y construido tres elementos fundamentales: varios tipos de hardware de robots móviles; una red de comunicaciones inalámbricas segura, rápida y confiable y un sistema de supervisión, que puede llevar al conjunto completo a desarrollar aplicaciones a distancia con control supervisorio de todo lo que sucede y posibilidad de actuar sobre él, en tiempo real.

Con los desarrollos que a continuación se exponen, y la presentación de los resultados, obtenidos en pruebas de campo y simulaciones, se pretende definir y comprobar la gran funcionalidad de estos diseños, que son eminentemente hardware; así como también hacer mucho hincapié en las estrategias desarrolladas, plasmadas en algoritmos de software, obteniendo controladores de diferente índole y concepción, especialmente matemática; estrategias que llevaran a efecto cosas que el hardware no podía conseguir por sí mismo p.e. la reducción del consumo de energía; estas estrategias – algoritmos han sido implementados en lenguajes de programación de alto nivel, como cumplimiento también de un objetivo primordial del proyecto de tesis.

En primer lugar se presenta la aplicación desarrollada, que aborda el objetivo específico de la reducción de los errores de odometría en robots móviles; en esta se utiliza un robot móvil omnidireccional, en conjunto con una cámara web, con los cuales se desarrolla un algoritmo que pone en operación un controlador de corrección de los errores que introducen los encoders de sus ruedas y las características del ambiente y diseño del robot; este algoritmo se basa en el uso del algoritmo genético y la descomposición de color, para resolver más rápidamente dicha corrección.

En segundo lugar, se presenta la aplicación desarrollada para cumplir varios objetivos planteados: el diseño e implementación de un sistema colaborativo entre varios robots de características heterogéneas, la búsqueda y localización de las posiciones relativas de los robots en el sistema y la implementación y pruebas de los sistemas de comunicaciones inalámbricas y supervisión. Aquí se desarrollan controladores para varios aspectos del sistema, que tiene la ventaja de ser capaz de escalar su operación desde tres a n robots heterogéneos.

Finalmente se presenta la aplicación, en la que se usa un robot manipulador móvil, que usa estrategias para realizar un menor consumo de energía en las tareas asignadas, este modelo usa las características cinemáticas y dinámicas de un robot móvil y un brazo robótico, las analiza de forma independiente y las integra en un solo controlador, de tal manera que, al aplicar un

modelo denominado ROC, logra reducir la cantidad de energía consumida por los motores en la consecución de la tarea.

6.1 PRIMERA APLICACIÓN: Reducción de los errores de odometría en robots móviles.

Para poder conseguir este objetivo, se realizan implementaciones de hardware y software montados sobre robots multisensoriales que puedan producir de forma objetiva la reducción de errores de odometría, al reproducir tareas de Path Planning o trayectorias diversas, esto aumenta en gran forma la confiabilidad en las tareas de localización y búsqueda.

En primer lugar, la selección del hardware: de entre los robots construidos, para poder comprobar las respuestas en sistemas robóticos actuales (y no redundar en un trabajo a lo mejor ya realizado en robots unicycle) se escoge un robot omnidireccional, y a partir de allí se implementa su multisensorialidad dotándole de una cámara web para producir un entorno de visión artificial que le permita medir, usando visión artificial, sus desplazamientos y corregir, usando un algoritmo en específico, el error introducido por odometría.

Para ello se ha diseñado y construido un sistema redundante de corrección de errores de odometría, sobre un robot omnidireccional, usando visión artificial y algoritmos genéticos.

Lo mencionado se centra en la implementación de un algoritmo para la corrección de errores producidos por la odometría de un robot omnidireccional en las tareas de locomoción autónoma, estos errores suelen ser producidos por falsas mediciones de los codificadores de las ruedas. El algoritmo de control se estructura integrando los datos de odometría y los valores de posicionamiento generados por la visión artificial. Se basa en el principio del algoritmo genético (hacer evolucionar una población de individuos sometiendo a acciones aleatorias) y utiliza señales de odometría, además del desplazamiento medido por una cámara web situada en la parte superior de la zona de interés. La propuesta del controlador se basa en el procesamiento de las imágenes que se hace a partir del modelo RGB basado en la síntesis aditiva, esto permite representar un color mezclando los tres colores primarios. El reconocimiento de la imagen facilita el detectar dos tonos y controlar cada imagen en un entorno estructurado; estas imágenes son analizadas por un algoritmo de reconocimiento de colores que determina los valores de posicionamiento en el plano y la orientación perseguida descrita en una propuesta de trayectoria definida. Este procedimiento se verifica experimentalmente para la evaluación del sistema con tareas de locomoción autónoma, con la construcción y prueba del robot multisensorial omnidireccional, y mediante el montaje del algoritmo sobre el robot.

La corrección de errores de odometría, no se resuelve únicamente colocando más y más sensores de todo tipo, la conjunción de su funcionamiento junto con sistemas inteligentes y algoritmos especializados son los que realmente resuelven el problema, aunque no al 100%. Es así como

existen técnicas de posicionamiento absoluto, como el emparejamiento de mapas, balizas y GPS, etc.; y otras técnicas relativas como la navegación por estimación, que se deriva de conceptos náuticos y aeronáuticos y tiene en cuenta las posiciones del vehículo en el rumbo y la velocidad, a partir de las cuales se determina su navegación instantánea. La odometría se basa en los mismos principios, ya que en cada iteración se refiere a la posición inicial del vehículo, y sus sensores estiman su desplazamiento y variación de orientación.

Los errores de posicionamiento en los robots móviles autónomos se describen o definen en dos grandes grupos: los errores sistemáticos, que consideran e incluyen las características físicas internas de construcción del robot, que difieren de sus diseños originales y causan diferentes respuestas a las acciones aplicadas; y los errores no sistemáticos, que provienen de la fuente común de error que es el entorno en el que opera el robot, es decir, el que interactúa con las ruedas y está en contacto directo con ellas; puede ser de naturaleza diferente, dependiendo del entorno y a menudo de la velocidad de movimiento.

Es necesario señalar que el conjunto de errores sistemáticos que se introducen en la odometría de un robot son a menudo imperceptibles, y también hay fallos en los controladores por el diseño y las concepciones de sus propiedades cinemáticas debido a la inexactitud de las piezas que lo conforman, como: radios de las ruedas, falta de alineación real sobre todo de las ruedas de tracción (al referirse a robots uniclo). Otros errores muy frecuentes son la incertidumbre de las distancias entre ejes causada por el contacto de la rueda con la superficie, la frecuencia real e ideal de trabajo de los codificadores, su resolución, etc.

Además, el entorno en el que se mueve el robot suele tener una incertidumbre total, lo que hace que sea completamente impredecible, ya que al no tener una certeza absoluta de los obstáculos, se producen giros inesperados y aceleraciones violentas o inconstantes, especialmente en suelos resbaladizos, que provocan lecturas erróneas de los encoders; o también es habitual registrar desplazamientos o giros sin haberlos realizado, cuando la superficie es irregular y el giro está en el aire instantáneamente, etc. lo que provoca errores no sistemáticos.

Se ha desarrollado un sistema que reduce considerablemente los errores de odometría de un robot móvil omnidireccional; se basa en el desarrollo e implementación de un algoritmo de control deducido a partir de una variación del algoritmo genético; este sistema usa principalmente un paradigma de bajo costo, que consiste en utilizar el sistema de menor costo económico para llevar a cabo la aplicación antes mencionada, aspecto que no influye de manera determinante en el logro de una gran estabilidad en el sistema de control, la adquisición de datos, el funcionamiento del controlador en tiempo real y la reducción tendencial hacia valores mínimos muy cercanos a cero o tolerables del error. La validación de los algoritmos de control de movimiento, estabilidad cinemática, control de error de rueda, algoritmo de corrección y

todos los componentes del sistema, se han realizado en el robot móvil multisensorial omnidireccional, que tiene comunicaciones en tiempo real con la plataforma de detección visual y la plataforma de control inteligente residente en una computadora remota. Este sistema de visión artificial se ha construido con el objetivo de determinar instantáneamente la posición y orientación del robot en todo momento con la base RGB; funciona dentro de un sistema de coordenadas fijas, lo que ha supuesto la construcción de un entorno estructurado de pruebas de movimiento y una detección del robot a través de la detección de dos puntos de color.

El término algoritmo genético se inspira en la selección natural y emula la evolución biológica. Partiendo de un grupo de soluciones, éstas se combinan para producir una descendencia, es decir, una nueva generación de "mejores soluciones". Estas nuevas soluciones se generan a partir de las antiguas, utilizando procedimientos de cruce y mutación. Al igual que se realiza en la naturaleza.

La naturaleza lleva a cabo intrínsecamente el proceso de evolución biológica de cada una de las especies, haciendo que se adapten a las condiciones en las que deben sobrevivir, encontrando individuos que presentan características especiales de adaptación, y de los que se heredan esas particularidades para que la especie en general pueda adaptarse. Para poder aplicar esta ciencia natural a un procedimiento de resolución de problemas de otra naturaleza, es necesario que se den una serie de condiciones especiales que la obliguen en cierto modo a conducirse hacia la solución; una de las principales condiciones es asegurarse de que la población sobre la que se va a aplicar el algoritmo tenga el tamaño inicial suficiente, de manera que se presente una amplia diversidad de soluciones probables de las que nazcan los individuos con mayor tendencia a encontrar la respuesta; el requisito de obtener una diversidad de alternativas significa que la generación de la población inicial debe ser necesariamente aleatoria; si no se utiliza este concepto, debe inferirse esta condición; el número de sujetos de respuesta generados debe ser tan alto como para garantizar la diversidad necesaria; esto se muestra en la Figura 6.1. Este principio se utiliza, teniendo en cuenta la idea de un algoritmo para crear la descendencia y seleccionar el mejor resultado (los mejores individuos) para la tarea de movimiento establecida dentro del robot, ya que hay que tener en cuenta que, sobre las muestras obtenidas en cada generación se aplican procedimientos de mutación y cruce, que hacen que las características de las soluciones mejoren en cada generación.

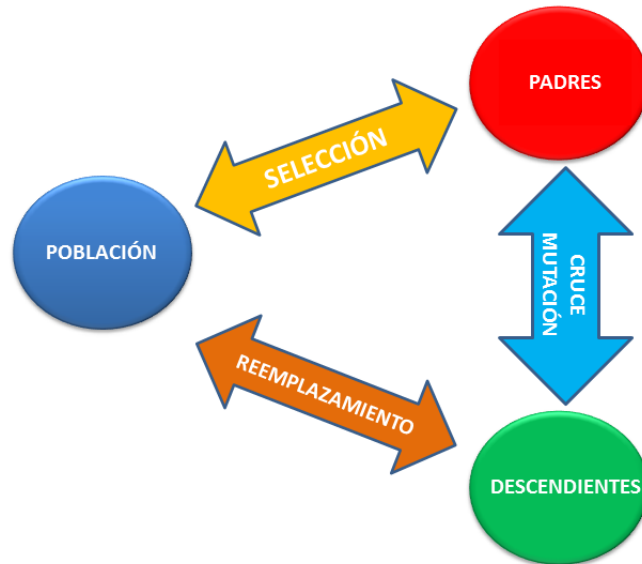


Figura 6.1. Consideraciones básicas de algoritmos genéticos.

6.1.1 Algoritmo propuesto.

Dado que el movimiento de un robot móvil de cualquier tipo genera errores, específicamente por el uso de la odometría y todo lo anterior, es crucial utilizar cualquier sistema que permita corregir estos errores. De la misma manera, ya se ha mencionado que hay muchas alternativas que añaden precisión a la odometría. Para ello, se utiliza principalmente un sistema de visión artificial, que transforma las imágenes obtenidas por la cámara en parámetros que el sistema de control de movimiento entiende para aplicar el algoritmo de corrección; estos parámetros de visión se transforman en unidades de longitud y se aplican directamente al sistema de posicionamiento instantáneo del robot.

La Figura 6.2 muestra el diagrama de bloques del sistema redundante para corregir los errores de odometría del robot omnidireccional. El bloque TRAYECTORIA DESEADA introduce un comando de trayectoria para el robot, es una orden con una curva matemática específica, con dimensiones establecidas que deben ser cumplidas; esto es realizado por el algoritmo de control y enviado desde la computadora principal a los motores del robot por medio de un sistema de comunicaciones inalámbricas. El movimiento del robot se mide por los dos mecanismos mencionados: la odometría, recogida por los codificadores de cada rueda omnidireccional; y por el sistema de visión artificial colocado en el exterior del robot a través de una webcam, instalada a una altura adecuada a todo el campo de movimiento del robot, para que no pierda detalle de sus movimientos (sistema de bajo coste). Este conjunto de datos se introduce en el algoritmo de corrección, modificando así el algoritmo de control para realizar movimientos que tengan mayor precisión y fidelidad a la trayectoria deseada. Este procedimiento se lleva a cabo teniendo en cuenta que, principalmente, el algoritmo de control basa su funcionamiento en la respuesta de

los codificadores y la fidelidad del mecanismo de retroalimentación del bucle de control PID, situado en los actuadores de cada rueda motriz del robot.

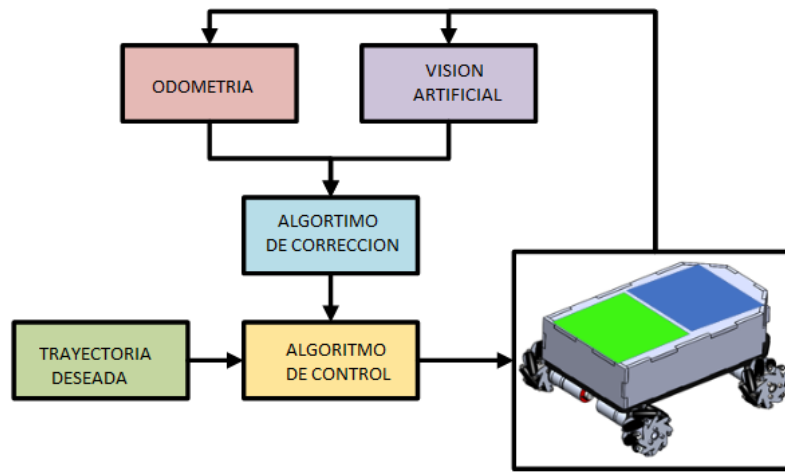


Figura 6.2. Diagrama de bloques del sistema de corrección de errores.

i: Odometría: Determinada por los datos obtenidos por los codificadores ópticos asociados a cada uno de los actuadores del robot, tras ejecutar un recorrido determinado por el robot omnidireccional. *ii:* Visión artificial: Captura una imagen del robot dentro del plano de visión mediante identificadores de color, que se superponen a él y son de colores explícitos, facilitando así su identificación dentro de un tono pastel del fondo general; mediante software se configuran los parámetros (resolución, modelo de color e intervalo), y con esta información se descomponen los componentes de color y luminancia; con estos valores y utilizando el modelo RGB y la síntesis aditiva de color, se determina la posición del robot en cada iteración del programa, es decir, en tiempo real; porque el tiempo de muestreo, es determinante en el sistema y entre todos los componentes involucrados. Una vez realizada esta detección, se utiliza el centroide de cada color, y de acuerdo con la resolución programada a la cámara, la información de los píxeles afectados se gradúa a la misma escala en la que han presentado los valores de la odometría. *iii:* Algoritmo de corrección: El principio de este algoritmo permite generar varias muestras entre los valores de posición instantáneos proporcionados por ambos métodos de medición, la odometría y la visión artificial, encuentra la diferencia entre ellos y los corrige mediante el uso del algoritmo genético. *iv:* TRAYECTORIA DESEADA: Este es la trayectoria establecida, que el robot omnidireccional debe recorrer en el plano de visión y el entorno antes mencionado en la sección *ii*; *v:* Algoritmo de control: Es el algoritmo a través del cual se controla el movimiento del robot, está basado en el modelo cinemático del robot, que se especifica a través de sus principales variables que son tres velocidades específicas: ω que representa la velocidad angular, y u_l y u_m que representan las componentes lineales de velocidad en ambos ejes dentro

del campo de coordenadas $\{R\}$ de X e Y ; la dirección de cada velocidad en la plataforma móvil se muestra en la Figura 6.3.

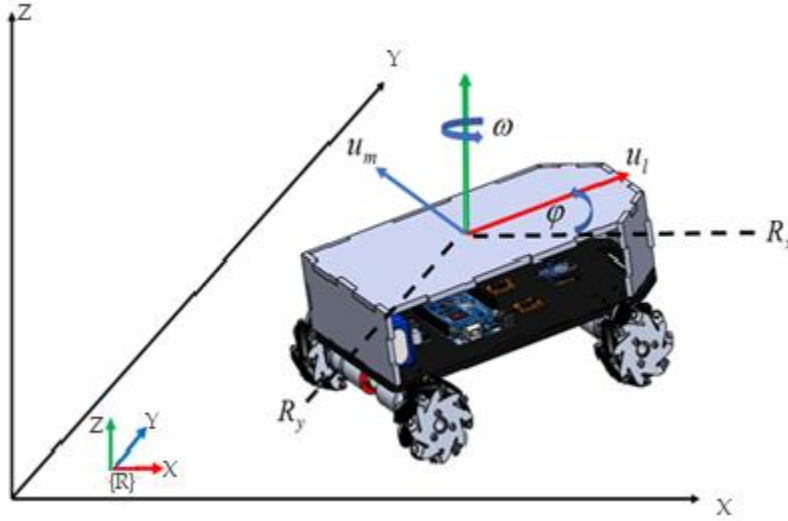


Figura 6.3. Estructura cinemática del robot omnidireccional.

Cada velocidad lineal está dirigida a uno de los ejes del plano $R(X, Y)$ y relacionada con el centro de gravedad del robot omnidireccional: u_l apunta a la dirección frontal, y u_m apunta a la dirección lateral izquierda. La velocidad angular ω hace que el sistema de referencia $\{R\}$ gire alrededor del eje Z .

La expresión matricial del comportamiento cinemático del robot dentro del marco inercial se define como:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_l \\ u_m \\ \omega \end{bmatrix} \quad (6.1)$$

Este sistema de ecuaciones representado (6.1) se expresa de manera más operativa utilizando la forma vectorial con la fórmula: $\dot{\mathbf{h}} = \mathbf{J}\mathbf{v}$ donde: \mathbf{h} es el vector de las posiciones en x, y, ϕ ; \mathbf{J} es la matriz jacobiana que contiene los parámetros cinemáticos del robot; y \mathbf{v} es el vector de las velocidades x, y, ϕ . Esto se refiere a la plataforma robótica completa, pero para hacer efectivo el movimiento, el punto de referencia, también llamado punto de control, es el centro de masa del propio robot.

Las dimensiones de configuración del robot, concebidas como un paso inicial en su diseño, se muestran en la Figura 6.4. Determinan la velocidad angular que debe desarrollarse en cada rueda, ya que aplicando un sistema de transformación, que se muestra a continuación, se obtienen estas velocidades independientes.

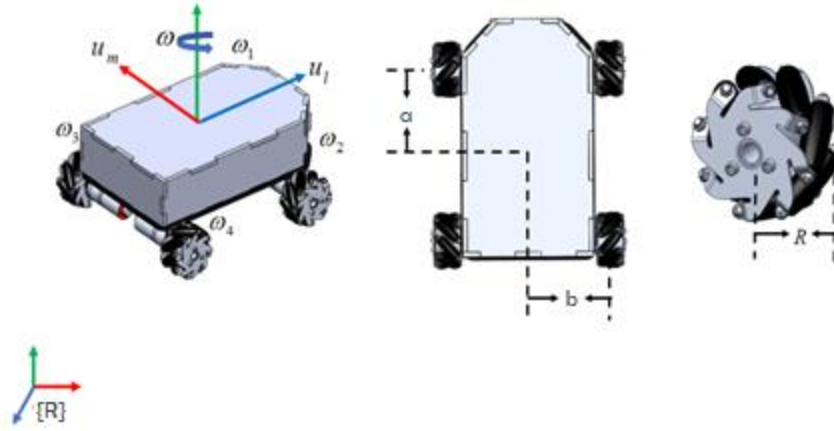


Figura 6.4. Dimensiones del robot omnidireccional a considerar en los modelos.

Como se había mencionado en el capítulo 2:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \frac{R}{4} \begin{bmatrix} 1 & -1 & -(a+b) \\ 1 & 1 & (a+b) \\ 1 & 1 & -(a+b) \\ 1 & -1 & (a+b) \end{bmatrix} \begin{bmatrix} u_l \\ u_m \\ \omega \end{bmatrix}$$

La expresión vectorial se expresa como:

$$\boldsymbol{\omega}_d = \mathbf{L}\mathbf{v}$$

En la que $\boldsymbol{\omega}_d$ representa el vector de las velocidades angulares de cada una de las ruedas omnidireccionales del robot, \mathbf{L} es una matriz de transformación de velocidad, obtenida de la cinemática inversa de las velocidades de cada una de las ruedas del robot omnidireccional, y \mathbf{v} corresponde a las velocidades lineales de movimiento de toda la plataforma del robot móvil omnidireccional.

El requisito fundamental de la funcionalidad del robot es que alcance la tarea programada de manera integral, es decir, el camino definido en tiempo real, esto se logra con el diseño adecuado de los controladores cinemáticos de la unidad robótica móvil, y la conveniente monitorización y sincronización de las comunicaciones; su funcionamiento se muestra más adelante. El robot debe seguir una trayectoria predeterminada que viene dada por una serie de puntos, que se consideran unidades de longitud, y que se muestrean en un intervalo de tiempo determinado. La ley de control establece que se necesita al menos una velocidad mínima para alcanzar la curva; esta velocidad se obtiene derivando la posición. Las posiciones del camino a seguir en un pequeño intervalo de tiempo están definidas por el vector $\mathbf{h}_d = [x_d y_d \phi_d]^T$, y la velocidad mínima a la que debe estar el vehículo es $\dot{\mathbf{h}}_d = [\dot{x}_d \dot{y}_d \dot{\phi}_d]^T$

Para diseñar el controlador cinemático del robot móvil omnidireccional, se parte de su modelo cinemático. La ley de control propuesta para su movimiento es:

$$\mathbf{v}_{ref} = \mathbf{J}^{-1}(\dot{\mathbf{h}}_d + \mathbf{K} \tanh(\tilde{\mathbf{h}})) \quad (6.2)$$

donde \mathbf{J}^{-1} contiene todos los parámetros cinemáticos del robot móvil omnidireccional, $\dot{\mathbf{h}}_d$ es el vector de las velocidades deseadas, y $\tilde{\mathbf{h}}$ es el vector de los errores de control en el que:

$$\tilde{\mathbf{h}} = \mathbf{h}_d - \mathbf{h} \quad (6.3)$$

Para limitar los errores probables y evitar cambios bruscos en el actuador, y como forma de filtrar y delimitar el error, se incluye un procedimiento analítico de saturación de las velocidades en el robot móvil omnidireccional, que se realiza utilizando la expresión $\tanh(\tilde{\mathbf{h}})$, que restringe el error en $\tilde{\mathbf{h}}$, para los límites superior e inferior, utilizando \mathbf{K} , que es una matriz de ganancia positiva definida para corregir los errores.

A efectos de estabilidad, la función definida positiva se establece $V(\tilde{\mathbf{h}}) = \frac{1}{2} \tilde{\mathbf{h}}^T \tilde{\mathbf{h}} > 0$ como la función candidata Lyapunov, y se obtiene la primera derivada en términos de tiempo:

$$V(\tilde{\mathbf{h}}) = \frac{1}{2} \tilde{\mathbf{h}}^T \tilde{\mathbf{h}} \quad (6.4)$$

$$\dot{V}(\tilde{\mathbf{h}}) = \frac{1}{2} (\dot{\tilde{\mathbf{h}}}^T \tilde{\mathbf{h}} + \tilde{\mathbf{h}}^T \dot{\tilde{\mathbf{h}}}) \quad (6.5)$$

$$\dot{V}(\tilde{\mathbf{h}}) = \tilde{\mathbf{h}}^T \dot{\tilde{\mathbf{h}}} \quad (6.6)$$

El mecanismo de retroalimentación del lazo de control se obtiene sustituyendo (6.2) en (6.1) de manera vectorial como se mencionó en los párrafos anteriores, de la siguiente manera:

$$\mathbf{v}_{ref} = V(\tilde{\mathbf{h}}) \quad (6.7)$$

$$\dot{\mathbf{h}} = \mathbf{J} \mathbf{J}^{-1}(\dot{\mathbf{h}}_d + \mathbf{K} \tanh(\tilde{\mathbf{h}})) \quad (6.8)$$

$$\dot{\mathbf{h}} = \dot{\mathbf{h}}_d + \mathbf{K} \tanh(\tilde{\mathbf{h}}) \quad (6.9)$$

Considerando:

$$\tilde{\dot{\mathbf{h}}} = \dot{\mathbf{h}}_d - \dot{\mathbf{h}} \quad (6.10)$$

se tiene:

$$\tilde{\dot{\mathbf{h}}} = \dot{\mathbf{h}}_d - \dot{\mathbf{h}}_d - \mathbf{K} \tanh(\tilde{\mathbf{h}}) \quad (6.11)$$

$$\tilde{\dot{\mathbf{h}}} = -\mathbf{K} \tanh(\tilde{\mathbf{h}}) \quad (6.12)$$

finalmente, reemplazando (6.12) en (6.6), se obtiene:

$$\dot{V}(\tilde{\mathbf{h}}) = -\tilde{\mathbf{h}}^T \mathbf{K} \tanh(\tilde{\mathbf{h}}) \quad (6.13)$$

Como se ha mencionado anteriormente, \mathbf{K} es una matriz definida positiva, para cualquier valor de $\tilde{\mathbf{h}}$, resultando siempre negativa; por lo tanto, el sistema cumple con el principio de estabilidad, según el criterio de Lyapunov.

Para describir el algoritmo de detección de visión artificial, se detalla que el sistema implementado obedece al esquema presentado en la Figura 6.5. Para ello se ha construido un entorno en el que se ha colocado una webcam a una altura de 3,4 m, de modo que el plano de visión (entorno de movimiento del robot) corresponde a un área rectangular de 2,56 m x 1,92 m.

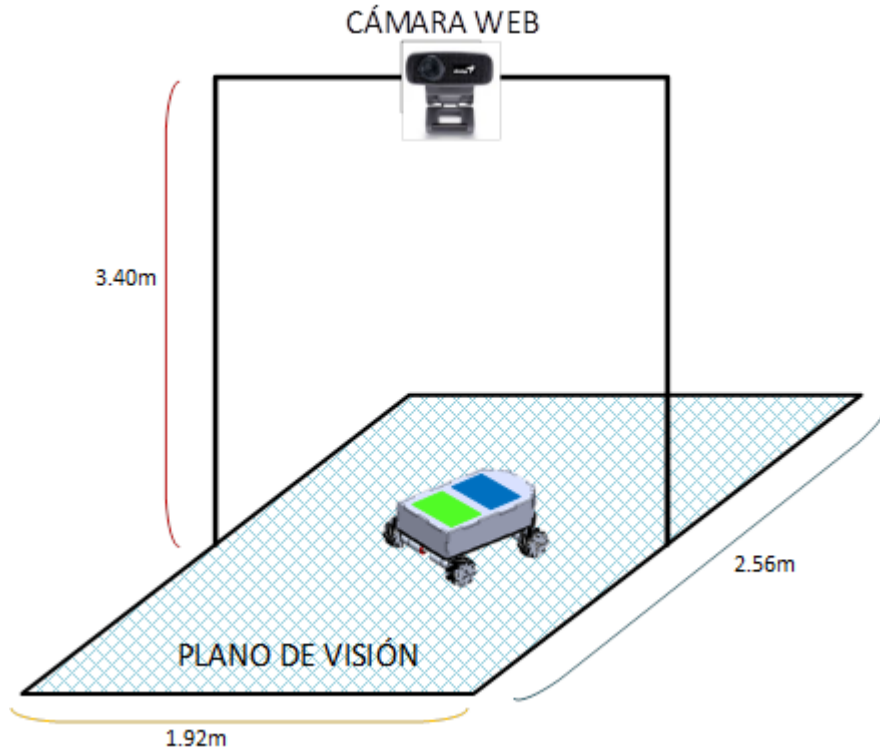


Figura 6.5. Estructura del plano de visión del sistema de corrección de errores.

Como se ha explicado anteriormente, el robot tiene dos identificadores de color en su parte superior que contrastan con el fondo del sistema de visión, lo que permite identificarlo en tiempo real. El procesamiento se realiza según el diagrama que se muestra en la Figura 6.6.

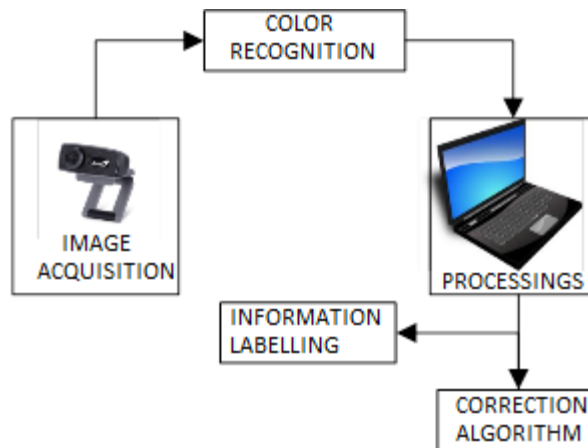


Figura 6.6. Diagrama de flujo del subproceso de visión.

El procedimiento comienza configurando el dispositivo de adquisición de imágenes (webcam), que es el dispositivo digitalizador de la información visual; configura la resolución y secuencia de las imágenes, y el modelo de color, en un intervalo entre imágenes previamente definido y que permite un correcto funcionamiento, éste es de 30 fps (fotogramas por segundo). Utilizando el modelo de color RGB, se puede descomponer cualquier espacio de color con su respectiva luminancia, por lo que este modelo matemático se utiliza para lograr la segmentación de color esperada mediante el análisis de la matriz de color. La forma de reconocimiento se obtiene a través de los dos identificadores verdes y azules, que se colocan encima del robot omnidireccional que se mueve sobre el fondo de color pastel del entorno físico construido, esta identificación se desarrolla siguiendo el procedimiento posterior:

- Crear e inicializar las variables para almacenar las coordenadas de cada color.
- Crear para almacenar las posiciones consecutivas de los identificadores de color.
- Delimitar la transparencia para la detección de cada color, este valor oscilará entre 0 y 1.
- Seleccione el dispositivo DAQ de la imagen y sus características.
- Analice cada tono de color identificado y relacione las propiedades entre sí.
- Asocie etiquetas de texto para las posiciones (x, y) de cada color y ángulo formado.
- Defina el centroide de la mancha de color.
- Identifique cada color individualmente.
- Asignar un cuadro a cada color.
- Transformar los datos de los píxeles en unidades de control (m).

Una vez obtenido el desplazamiento por ambos métodos (odometría y sistema de visión artificial) de la plataforma omnidireccional, es conveniente referenciarlos en el mismo sistema, utilizando un software de alto nivel, y de esta forma es posible visualizar inicialmente la

diferencia entre ambos métodos, para trayectorias definidas, en las que sus particularidades introducen diferentes errores característicos. En las figuras presentadas en la sección 4 se pueden ver los resultados de este proceso y la marcada diferencia entre los métodos.

Considerando que uno de los aspectos del desarrollo actual de la inteligencia artificial corresponde a la técnica de los algoritmos genéticos, que se basan en la propia naturaleza, en cuanto al hecho de producir siempre de manera primitiva o modificada genéticamente individuos mejores adaptados a características específicas, las nuevas generaciones se basan en los atributos de las más antiguas, que naturalmente tendían a desarrollarse hacia tal o cual aspecto específico; además de ser intrínsecamente paralelas. El procedimiento desarrollado es el de la mutación o el cruce para la producción de nuevas especies. De esta manera, esta técnica se utiliza para combinar los mejores resultados del algoritmo que producen trayectorias cada vez mejores, y así reducir enormemente los errores y a una velocidad bastante alta, porque la dinámica del robot lo exige.

Para la implementación del algoritmo que corrige los errores de odometría del robot, se utilizan como parámetros iniciales de población los valores obtenidos por los métodos mencionados y el principio de selección natural; luego se seleccionan las muestras obtenidas por cada sistema para la posición en X , la posición en Y y la orientación ψ , de tal manera que cada par de datos genera varios valores (cuatro) de descenso con una probabilidad diferente y aleatoria y al mismo tiempo equidistante; de manera que se cumpla con la reducción de los errores en las trayectorias; a partir de ellos se deben elegir las mejores soluciones, esto se muestra en la Figura 6.7.



Figura 6.7. Criterios de selección de los descendientes, en el algoritmo genético.

La terminología utilizada en los algoritmos genéticos incluye términos de reproducción natural y animal, así también, cromosomas, mutación, evolución o cruce, lo que equivale en nuestro sistema a mencionar las soluciones generadas como puntos de trayectoria, y ellas mismas con ciertas modificaciones de probabilidad y otras características, que hacen que el método se aplique efectivamente en la solución final del sistema redundante.

Básicamente, un algoritmo genético evalúa la puntuación de cada cromosoma generado y permite que, en base a esas características o a algunas ya mejoradas, se sigan reproduciendo de tal forma que los resultados más adecuados tengan más probabilidad de reproducirse, mutando un nuevo gen en cada generación y aumentando la población que volverá a participar en el proceso hasta encontrar la condición de finalización, que será en nuestro caso la reducción a cero del error de desplazamiento del robot. El tamaño de la población inicial resulta ser un factor determinante, ya que, si es insuficiente, hay muy pocas probabilidades de que la reproducción sea efectiva; y si es excesiva, la operación sería extremadamente lenta, por lo que hay pruebas que determinan límites para definir lo ineficiente que es aumentar el tamaño de la población, ya que no es posible de ninguna manera aumentar la velocidad de resolución del problema.

Además, hay dos parámetros muy necesarios de considerar: son la probabilidad de cruce y la probabilidad de mutación; el primero, es un parámetro de frecuencia, que da la certeza de que se producirán diferentes cromosomas de tal manera que los hijos no tengan un 100% de las características de los padres, sino un cruce que derive en mejores soluciones; y el segundo, también un parámetro de frecuencia, que determina la posibilidad de que el cromosoma mantenga sus características o que las cambie completamente.

La aplicación de los operadores genéticos se realiza mediante la codificación de una serie de variables dentro del cromosoma y se aplican a una gran parte de la población en cada iteración, deteniéndose cuando llega a la solución óptima que más se aproxima a la trayectoria real del robot. Para la selección, los tres procesos genéticos (mutación, cruce y evolución) son simulados por tres grupos, clasificados en orden ascendente de probabilidad y en pares, donde las posiciones representan: Mutación = 1 y 2; Cruce entre 3 y 4; y Evolución entre 5 y 6, como se muestra en la Tabla 6.1.

Tabla 6.1. Valores de probabilidad para la aplicación del algoritmo genético.

	POSICIONES DEL VECTOR	PROBABILIDADES	
MUTACION	1	$0 \leq P < 0.16$	VISION
	2	$0.16 \leq P < 0.32$	
CRUCE	3	$0.32 \leq P < 0.48$	
	4	$0.48 \leq P < 0.64$	
EVOLUCION	5	$0.64 \leq P < 0.83$	
	6	$0.83 \leq P \leq 1$	ODOMETRIA

Así es como el algoritmo de corrección utiliza la información de los dos sistemas de medición de la trayectoria y el desplazamiento del robot para hacer su corrección; el controlador recibe en tiempo real las señales de aproximación en las que se integran la odometría y el sistema de visión.

6.1.2 Implementación.

Como se había mencionado, la implementación de todo lo anteriormente descrito se hace en un robot móvil omnidireccional construido para el efecto y que presenta las características que se observan en la Figura 6.8.; permite una gran versatilidad de movimiento, ya que tiene capacidad de desplazamiento simultáneo en los planos XY, y su rotación con el plano Z.

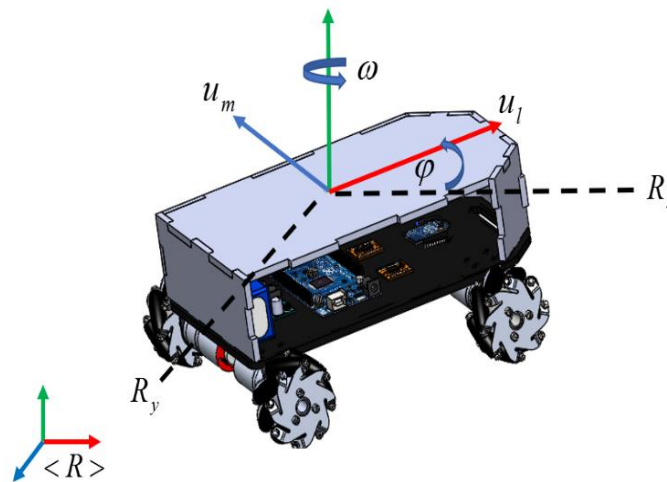


Figura 6.8. DUT robot omnidireccional.

Basado en todo lo que se escribió y desarrolló en el capítulo II de este trabajo, el diseño se centra principalmente en sus características omnidireccionales y la necesidad de su velocidad moderada, además de su forma autónoma de desplazamiento y control inteligente a distancia y con una gran capacidad de control en tiempo real, por lo anterior, está compuesto por seis bloques funcionales fundamentales, los cuales se explican a continuación y se pueden ver en la Figura 6.9, aquí se describen las características de cada etapa y su función minimalista, lo que también nos lleva a definirla como una plataforma de bajo costo económico y moderado costo computacional y de reducido consumo de energía también; se muestran las etapas operativas, denominadas: energización, control, posición, actuadores, comunicación e interfase.

a. Bloque de energización.

Como en la mayoría de los dispositivos móviles y de radiocontrol, el sistema de desplazamiento omnidireccional del robot móvil está alimentado por una batería de polímero de litio LIPO, que ha sido seleccionada por su versatilidad en cuanto a sus dimensiones, peso y capacidad de carga, tiene una corriente nominal de 2,2 A, suficiente para establecer un control dinámico del

movimiento, este suministro de tensión no se alimenta directamente a las piezas del robot, sino a través de un regulador de tensión, que regula la tensión de CC exactamente a 6V, requerido por la tarjeta de control de comunicaciones y la movilidad del robot.

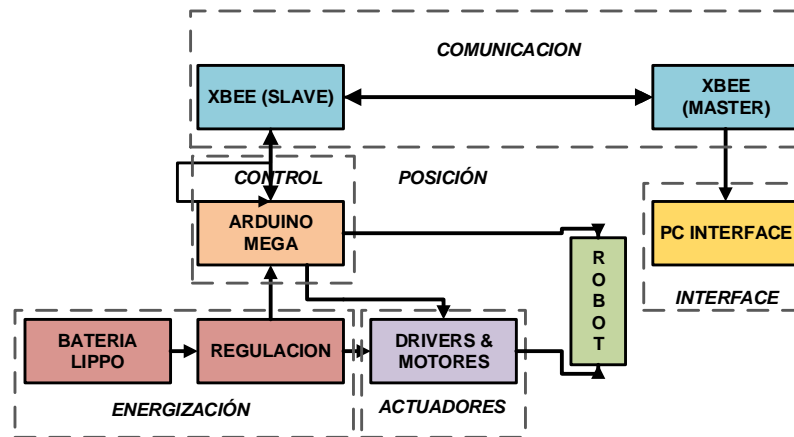


Figura 6.9. Hardware de implementación del algoritmo de reducción de errores

b. Bloque de control.

Este es el sistema encargado de controlar el movimiento omnidireccional del robot y se basa en una placa Arduino Mega 2560, que recibe las señales del ordenador central que ejecuta el algoritmo de control de velocidad de cada rueda para obtener la trayectoria final del robot en tiempo real; interpreta las señales procedentes de este sistema que se caracterizan por los controles PID a cada una de las ruedas, es decir a sus motores centrales.

En la Figura 6.10 se presentan los bloques constitutivos del sistema de control cinemático, en los que se observa el bloque individual de la compensación dinámica necesaria para reducir el error de velocidad, éste recibe como entradas las velocidades angulares deseadas, calculadas por el controlador cinemático, obteniendo la velocidad angular del robot; al compensar dinámicamente el movimiento del robot, se permite que sus velocidades de desplazamiento varíen, y se permite transportar de manera controlada objetos que introducen mayor peso, el error de velocidad motiva el proceso de compensación dinámica.

c. Bloque de actuadores.

Para ello se utilizan motores de corriente continua que funcionan con señales de entrada PWM procedentes del controlador y enviadas a través del sistema de comunicaciones para ser ejecutadas por el controlador interno. Es un sistema que integra 64 encoders CPR y con un par de 8 Kg-cm; este diseño permite que el robot se mueva según la trayectoria sin medir el peso que se le aplica, una vez que permite romper la inercia del sistema completo.

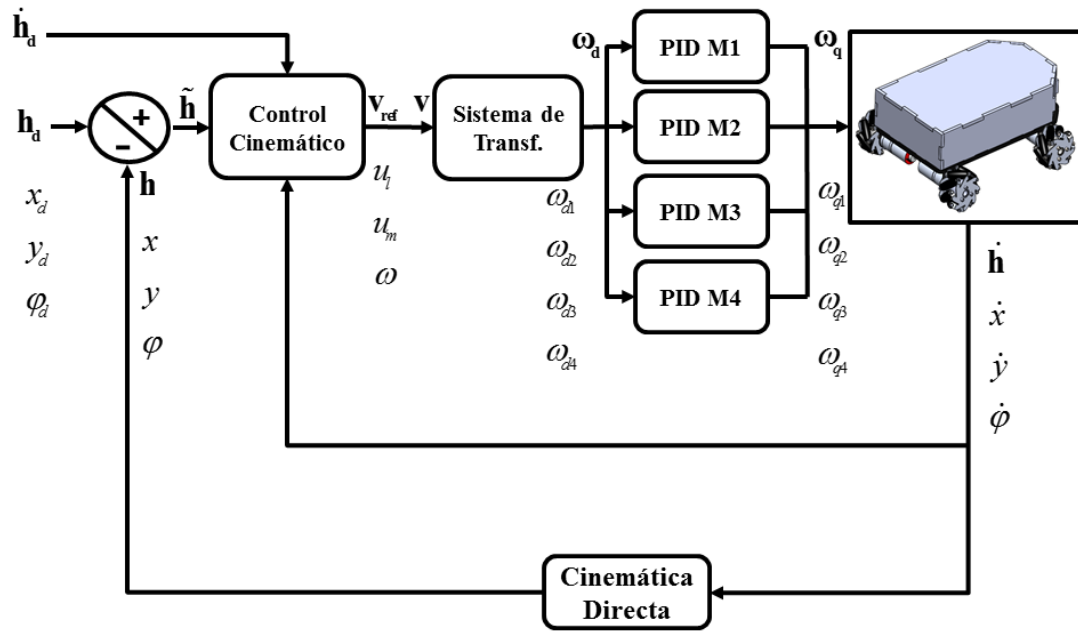


Figura 6.10. Diagrama de bloques del sistema de control de movimiento del robot omnidireccional.

d. Bloque de comunicaciones.

Como ya se ha mencionado, el funcionamiento del sistema en su conjunto es en tiempo real, y esta característica se menciona porque no todo el sistema está montado en el robot, al contrario, los algoritmos más complejos se implementan en una computadora remota, que envía la información a través de una red inalámbrica.

Considerando todas las características de conexión, diseño, uso de banda libre, transferencia de datos, número de bits, uso de energía, etc. se decidió utilizar una red comercial ZigBee para conectarse de forma inalámbrica al ordenador central y al robot; el módulo de la plataforma central está configurado como coordinador y también tiene las funciones de controlador de red y selección de velocidad del programa escrito en MATLAB, en el otro lado de la comunicación está el módulo del robot, configurado como dispositivo final. La transferencia de datos de las dos plataformas se realiza de forma interactiva haciendo que el ordenador entregue las señales de control de movimiento y las señales de corrección de errores de odometría, y que el robot entregue las señales de sus codificadores para la interpretación de su desplazamiento. Esto se hace introduciendo variables booleanas -que sincronizan la comunicación- en el flujo de datos, permitiendo la precisión de los datos y evitando superposiciones que afectarían a la corrección del error de odometría.

e. Bloque de Interfase Gráfica de Usuario.

Al existir una computadora que envía señales de control desde más de dos sistemas fundamentales, y al responder el robot remoto a estas señales, la interfaz gráfica desarrollada en la pantalla de la computadora se vuelve más intuitiva, convirtiéndose en una HMI (Human Machine Interface), ya que el operador, con la introducción de los parámetros de movimiento en la interfaz, ordena el movimiento símil del robot, y cuando corrige los errores de su desplazamiento, aumenta la efectividad en la medición de sus trayectorias.

La necesidad de escribir los códigos de los algoritmos de control y corrección se resuelve con el uso de MATLAB debido a su intuición, facilidad de programación y al amplio conjunto de funciones matemáticas que posee, privilegiándolo sobre otro tipo de software de alto nivel. Es por esta razón que la interfaz se desarrolla también en este software, a pesar de que se puede hacer en tiempo real y con mayores prestaciones con otra aplicación y el uso de la comunicación DDE. Desde esta interfaz es posible operar la plataforma completa: habilitando las comunicaciones, observando el estado de la misma en tiempo real y los controles necesarios para introducir las velocidades del sistema según una trayectoria propuesta, de forma sencilla y minimalista para el usuario; su diseño se muestra en la Figura 6.11.

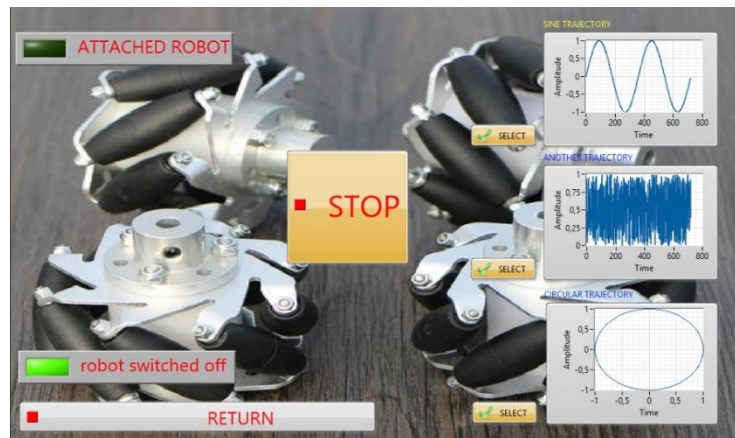


Figura 6.11. Interfase de operador.

Con la ejecución de esta interfaz, el programa envía los datos de las velocidades requeridas por cada motor para llevar a cabo el movimiento programado del robot, esto se hace dentro de una trama de comunicaciones que se ejecuta en un ciclo de tiempo específico; después de lo cual el robot, vía Wireless, devuelve los datos reales de su movimiento, presentando el resultado del comportamiento de todas las variables que intervienen en el sistema, en curvas para su visualización y análisis. Estas curvas que se visualizan ya reflejan los resultados experimentales del movimiento del robot.

6.1.3 Resultados experimentales.

Con el fin de verificar la fidelidad de los movimientos del robot al entorno ordenado para su ejecución, se realizaron pruebas con diversas formas geométricas normales; y a través de ellas, verificar la eficacia del algoritmo en el controlador del movimiento omnidireccional del robot móvil. De esta manera, según lo que se observa en la interfaz de usuario, se establece la posibilidad de realizar las pruebas sobre una señal sinusoidal, así como sobre una circunferencia y sobre una curva definida por una ecuación particular; cada una de ellas definiendo previamente las condiciones de funcionamiento, concretamente el tiempo de muestreo, y los valores de las ganancias; también se introduce en los experimentos el valor de la probabilidad del algoritmo genético que presenta los mejores resultados de seguimiento para la corrección.

Las figuras siguientes representan una secuencia de los resultados de la aplicación de dos funciones al movimiento del robot; en la Figura 6.12 una función circular con el valor de probabilidad exacto en el algoritmo genético y en la Figura 6.13 la simulación del movimiento, confrontando la propuesta de trayectoria y la respuesta del algoritmo.

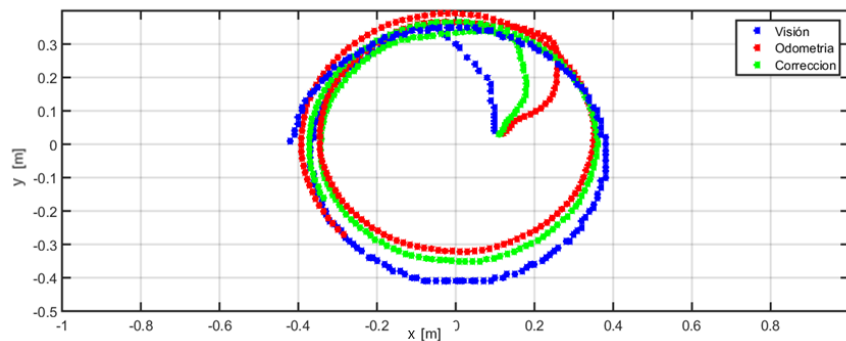


Figura 6.12. Camino circular descrito por el algoritmo de corrección.

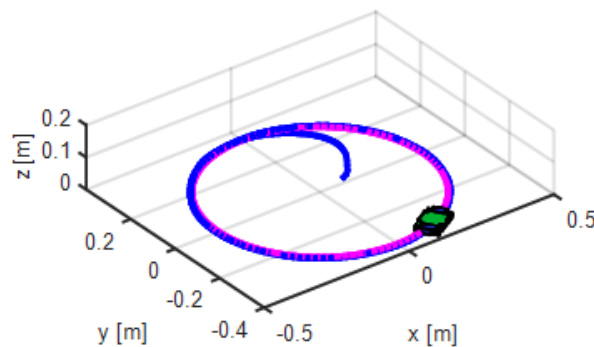


Figura 6.13. Simulación del control en la trayectoria descrita

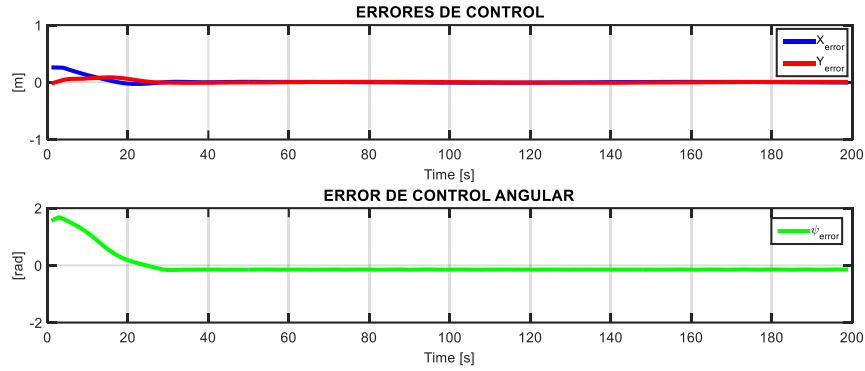


Figura 6.14. Errores al aplicar el algoritmo de corrección.

Se puede ver en la Figura 6.14 que los errores de control, que inicialmente tiene un valor determinado pero que luego se reducen a cero, como resultado del cumplimiento de la tarea del controlador; también se puede ver en la Figura 6.15 las acciones de control sobre el robot en su velocidad frontal, velocidad lateral y acción de control angular, los valores se encuentran se encuentran en el rango permitido para acciones en un robot móvil.

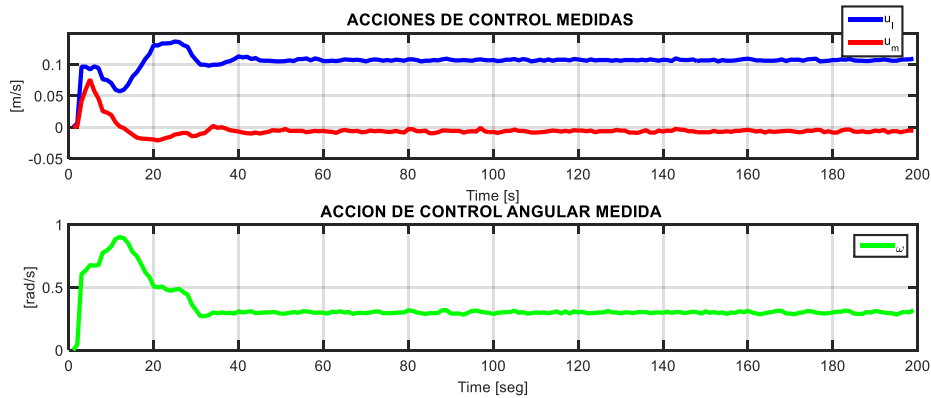


Figura 6.15. Acciones de control para reducir los errores de velocidad angular y lineal.

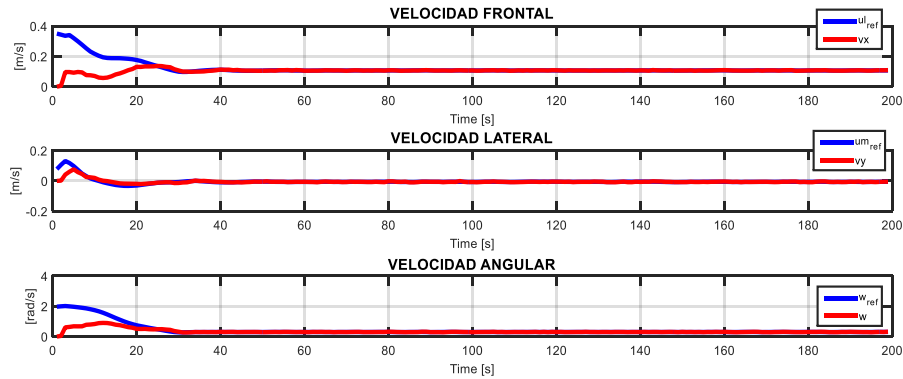


Figura 6.16. Acciones de control en las componentes de velocidad del robot.

La Figura 6.16 muestra la respuesta en velocidad del robot, y se observa la relación entre la señal de referencia con respecto a la señal medida, se puede verificar aquí el correcto seguimiento del controlador, se verifica que el tiempo de respuesta del controlador, también es muy aceptable.

De la misma manera, se presentan los resultados de la aplicación del experimento a una geometría sinusoidal, las respuestas pueden verificarse en las figuras siguientes. En la Figura 6.17 se presentan las tres señales de interés, y se observa que nuevamente, como en el caso anterior el controlador aplicado a una probabilidad exacta, tiende hacia la señal que supone presenta cero error, pero como la señal de visión presenta el error existente lo corrige, justamente con la tendencia resultante de la probabilidad aplicada en el algoritmo genético, se observa la tendencia de reducción del error con la corrección, esta no presenta el valor promedio de las dos mediciones sino el valor siguiente más adecuado a la probabilidad aplicada en el algoritmo; aunque la señal de visión tiene en este caso (y por las características de la cámara) menos fidelidad, la corrección trata de llevar a la señal al sistema en donde se producen menos errores, que sería la señal de odometría, tal como se presentaba la Tabla 6.2, lo que significa que proporcionándole más resolución a la cámara la corrección será mucho más efectiva.

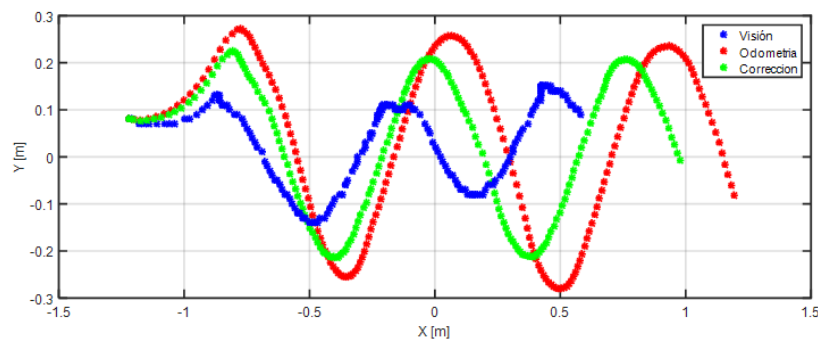


Figura 6.17. Operación del algoritmo de corrección para una forma de onda sinusoidal.

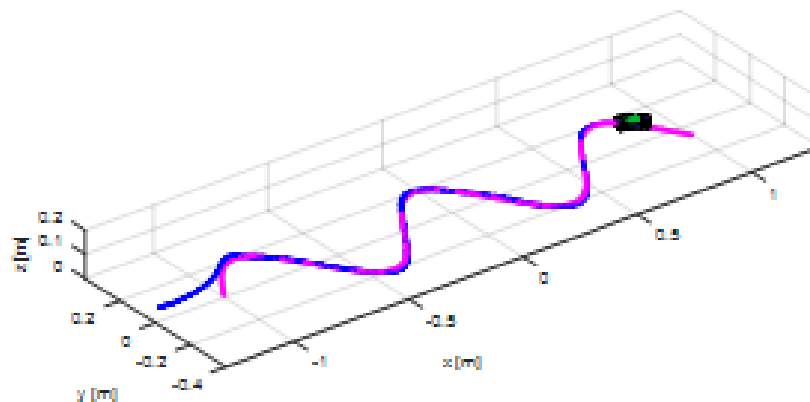


Figura 6.18. Simulación del algoritmo de corrección, aplicada una función sinusoidal.

En la Figura 6.18 se muestra también la simulación de esta trayectoria y la respuesta del controlador; se muestra el desplazamiento del robot utilizando los valores de corrección con respecto a la curva sinusoidal que tiene que describir, en este caso el valor de probabilidad utilizado hace que prácticamente se superponga la señal simulada (respuesta del controlador) con la señal propuesta; en la Figura 6.19 se visualiza la corrección del ángulo de giro del robot, nuevamente con el valor de probabilidad apropiado, se observa que no es el promedio de las dos mediciones sino la aplicación del algoritmo genético con tendencia a donde se considera introduce menor error.

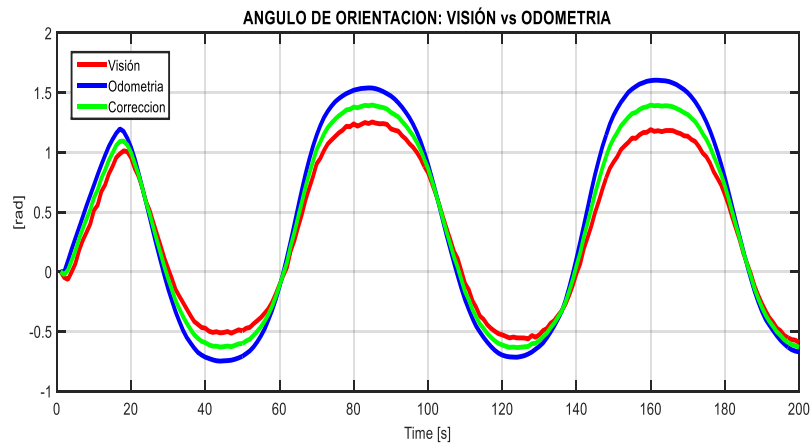


Figura 6.19. Orientación de la corrección del algoritmo a una señal sinusoidal.

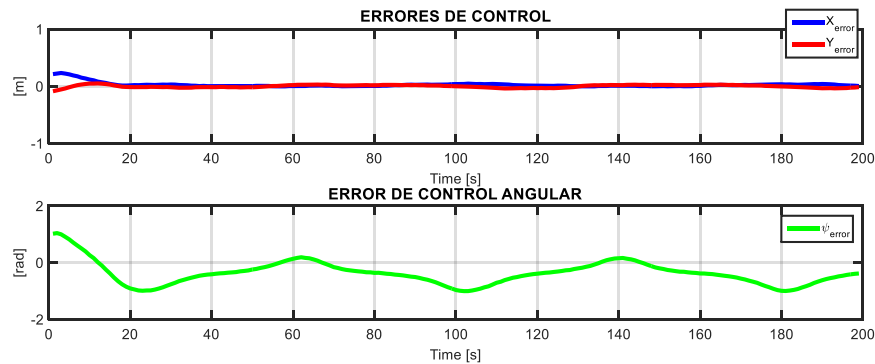


Figura 6.20. Errores de control para la señal sinusoidal.

En la Figura 6.20, al igual que en caso anterior, ya para esta geometría en particular, se presentan los errores de control en x , y , ϕ se verifica nuevamente la tendencia a cero de estos errores en tiempos adecuados.

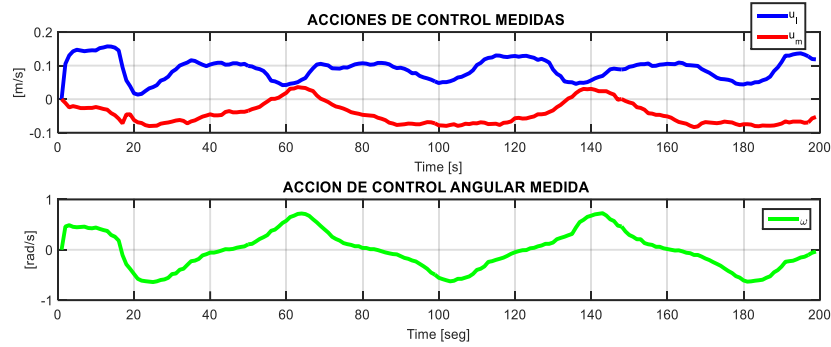


Figura 6.21. Acciones de control aplicadas para una trayectoria sinusoidal.

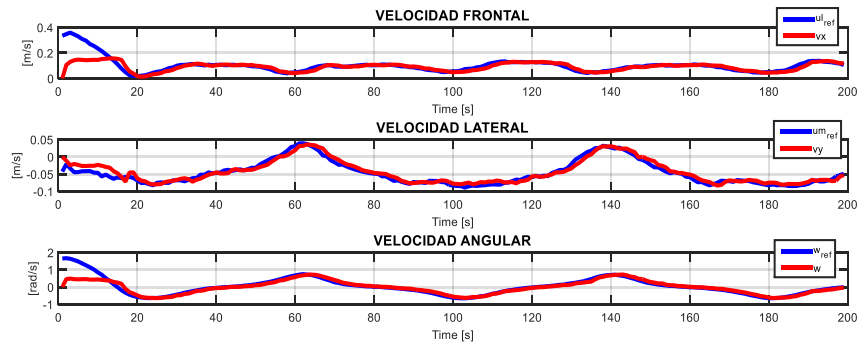


Figura 6.22. Acciones de control aplicadas a los componentes de velocidad del robot cuando se emplea una señal sinusoidal.

Finalmente, la Figura 6.21 muestra las acciones de control para el robot, se muestra la acción de control sobre la velocidad frontal, velocidad lateral y la acción de control angular; nuevamente se verifica que dichos valores están dentro de un rango muy aceptable para un robot móvil; La Figura 6.22 muestra de forma gráfica la señal de referencia con respecto a la señal medida de las velocidades frontal, lateral y angular del robot.

Como se puede observar, la reducción de los errores de odometría (la técnica más común y simple de medición de trayectorias de un robot móvil) no se basa en la utilización solamente de un hardware muy especializado y de múltiples sensores (por no decir todos los existentes) colocados en él; tampoco de un algoritmo completamente inteligente y experto, que no use casi ninguna prestación de una plataforma de hardware para que esta se mueva sin errores, ninguna de las dos, sino, más bien de una combinación de todo lo que se ha expuesto en este numeral: la plataforma de hardware construida bajo especificaciones y dimensiones establecidas en diseño, su modelación cinemática acorde a este diseño y prototipo implementado, su compensación dinámica, (que es uno de los aspectos determinantes que hacen que en sí la odometría produzca menos errores); también un sistema que considere la mayor parte de aspectos del ambiente o factores externos que puedan influir; la elección de componentes electrónicos de alta calidad, el

establecimiento de una red de comunicaciones excelente, el desarrollo de los algoritmos de control de forma efectiva, etc. Hacen que la aplicación consiga su fin, así como este trabajo de tesis también; es decir se remite nuevamente a repetir que no existe hardware ideal que desarrolle aplicaciones ideales, es la combinación de todo lo mencionado que cumple con los objetivos.

6.2 SEGUNDA APLICACIÓN: Control heterogéneo de robots considerando objetivos secundarios.

En esta aplicación se analiza el cumplimiento de los objetivos primarios y secundarios de un conjunto de robots de características heterogéneas y con diferentes modos de control y operación, trabajando de forma cooperativa, asumiendo que la posición de cada robot, el camino a seguir y la forma de las formaciones se conocen como la condición inicial, teniendo en cuenta al mismo tiempo que el conjunto robótico conoce el entorno a partir de información externa o de datos previamente almacenados. Mediante la propuesta de un algoritmo de control basado en el álgebra lineal, que abarata el costo computacional y se hace fácilmente programable, que son una gran ventaja en la implementación del sistema, el conjunto de robots puede realizar una tarea de forma coordinada siguiendo un camino deseado como tarea primaria, mientras que la formación de los mismos se altera dependiendo del entorno o de los requerimientos de la misión (tareas secundarias); esta es la principal novedad del sistema propuesto. A diferencia de las formas tradicionales de controlar un grupo de robots terrestres, en este trabajo el conjunto robótico se mueve a lo largo de un camino con una velocidad deseada no parametrizada en el tiempo, que puede depender de varios propósitos (evasión de obstáculos encontrados en el camino, optimización de la energía, etc.), mientras que simultáneamente puede alterar la formación del conjunto mediante la modificación de las distancias y orientaciones de las proyecciones principales y secundarias. A diferencia de otros trabajos, el esquema de formación de los robots heterogéneos se analiza a partir de proyecciones entre un par de robots (ya sean físicos o virtuales, según los requerimientos), lo que proporciona flexibilidad y escalabilidad a la hora de sumar los sistemas robóticos (o sea incrementarlos en el sistema). Asimismo, se diseña, simula e implementa el controlador basado en el álgebra lineal sobre el conjunto de robots heterogéneos, analizando además su estabilidad. Otra gran ventaja del sistema propuesto es que al ser un conjunto de robots heterogéneos los que operan, hace que el control cinemático y dinámico de cada uno de ellos resida en ellos, reduciendo así la carga de procesamiento del sistema y dando la posibilidad de aumentar los robots de cualquier característica operativa, incluyendo su control cinemático y dinámico en el mismo, sin necesidad de modificar ninguna otra parte del sistema en absoluto. No se conoce ningún trabajo que haya hecho algo similar. El controlador propuesto utiliza una estructura de control jerárquica, con el fin de proporcionar

escalabilidad al sistema y, al mismo tiempo, no saturar la unidad de procesamiento, fusionando así los procesadores de información centralizados y descentralizados. En un nivel jerárquico muy alto, una computadora centralizada se encarga de generar las acciones de control para lograr los objetivos primarios y secundarios, mientras que, a nivel local, cada miembro del conjunto heterogéneo de robots incluye su propia unidad de procesamiento para lograr un control cinemático y dinámico y también para proporcionar retroalimentación de la información de odometría, que se comparte mediante comunicaciones inalámbricas basadas en un marco de datos (suponiendo que no haya limitaciones dadas por el entorno). Además, otra ventaja relevante de la propuesta es el bajo costo computacional obtenido mediante el método de control basado en el álgebra lineal, evitando la saturación tanto de la computadora centralizada como de las ubicadas en cada robot. Hay trabajos basados en las teorías del Álgebra Lineal para el control colaborativo de robots móviles, centrados en la técnica de leading-follower; sin embargo, allí no se considera la escalabilidad de los robots ni la ejecución de los objetivos de control secundarios, es decir, la modificación de la formación de los robots y la evitación de obstáculos, entre otros criterios. Nosotros, en nuestro trabajo, hemos hecho todo esto, lo que lo convierte en una mejora sustancial.

6.2.1 Esquema de control Multicapa.

El esquema multicapa define cómo se distribuyen las variables apropiadas que conforman el control cooperativo de los robots móviles heterogéneos (Figura 6.23). Teniendo en cuenta que cada una de las capas tiene diferentes funcionalidades, el esquema se divide en seis secciones: (1) La adquisición de las características del entorno permite planificar el camino a seguir, manteniendo la integridad de los sistemas robóticos, así como cumplir la tarea preprogramada. Esta sección tiene dos capas, y en la capa de planificación fuera de línea se configuran los parámetros iniciales, es decir, la trayectoria deseada a seguir, las ubicaciones de partida de todos los sistemas robóticos y la estructura de formación deseada. La capa de planificación en línea permite la ejecución de los objetivos secundarios, reaccionando a los cambios ambientales imprevistos en la capa off-line, producidos por la falta de espacio para cumplir los parámetros de formación deseados, los objetos a evitar a lo largo del camino, las condiciones ambientales que requieren cambiar la formación realizada, entre otras condiciones, que pueden alterar el cumplimiento de la tarea. Esto también es muy importante porque, a través de este esquema, se puede navegar entre los ambientes estructurados y no estructurados. Cuando se conoce el camino, o el entorno operativo ya ha sido obtenido previamente, esta información se introduce en esta capa, si no, y como los robots son multisensoriales, se puede instalar fácilmente un radar y hacer la planificación del camino y seguir esas trayectorias encontradas en tiempo real.

El objetivo de la capa de control de la formación (2) se centra en el cálculo de las acciones de control para que los vehículos mantengan un posicionamiento que satisfaga la formación

deseada, generando las señales de control para cada uno de los robots. La (3) capa de control cinemático toma como referencia las velocidades generadas por el control de la formación (nivel superior), es decir, los controladores locales reciben velocidades de referencia para seguir el perfil guiado por la formación, por lo que estos controladores son responsables de entregar velocidades de maniobrabilidad que corresponden a cada robot heterogéneo involucrado (omnidireccional, tipo coche, unicycle). La (4) dinámica de cada robot que actúa como mecanismo individual se resuelve mediante la capa de compensación dinámica adaptativa; dentro del esquema de control y operación esta capa es de suma importancia y proporciona, como se mencionó anteriormente, la ventaja de que si se incrementan los robots de otras características, cada uno tiene su propio controlador en la parte superior, lo que significa que la única modificación importante que hay que hacer al sistema es introducir los datos de este robot en el marco de las comunicaciones, eliminando completamente cualquier cambio fuera de éste. La capa de (5) Robots representa el conjunto de robots heterogéneos considerados en la formación. Además, se utilizan dos robots virtuales para el cálculo de cada una de las proyecciones. Por último, la capa (6) entorno representa el espacio estructural donde el robot interactúa, considerando los obstáculos y otros objetos físicos.

De lo que se ha descrito, puede observarse que el trabajo que presentamos, difiere de la literatura de control en colaboración en los siguientes puntos: 1) Propuesta de un esquema de control modular de varias capas, es decir, es posible aplicar diferentes técnicas de control sin modificar la estructura general del control en colaboración entre n robots; 2) el esquema de control considera tanto la técnica de control centralizado como la técnica de control descentralizado, que permite generalizar el control en colaboración a n robots heterogéneos; 3) los algoritmos de control propuestos se basan en la teoría del álgebra lineal, que tiene como ventaja un bajo costo de cálculo y, por lo tanto, puede aplicarse en robots construidos con tecnología de bajo costo. Además, al considerar la técnica del álgebra lineal para el diseño de los algoritmos de control, es posible incorporar la redundancia en las matrices de transformación de la formación, por lo que es factible ejecutar la tarea principal (seguimiento del camino de las tareas predefinidas) y varias tareas secundarias, por ejemplo, modificación de la formación, evasión de obstáculos, menos movimientos, entre otros criterios de control; 4) el algoritmo propuesto considera el seguimiento del camino, es decir, seguir el perfil deseado sin ser parametrizado en el tiempo. La mayoría de los trabajos de la literatura realizan el seguimiento de la trayectoria, lo cual no es factible de implementar en aplicaciones reales ya que el seguimiento de la trayectoria define la posición deseada en un momento determinado. Es decir, tiene un perfil de rastreo parametrizado en el tiempo; y finalmente 5) en este trabajo hay un análisis formal de la estabilidad de cada uno de los algoritmos de control propuestos, en el que se demuestra matemáticamente la estabilidad asintótica, es decir, que los errores de control tienden a cero.

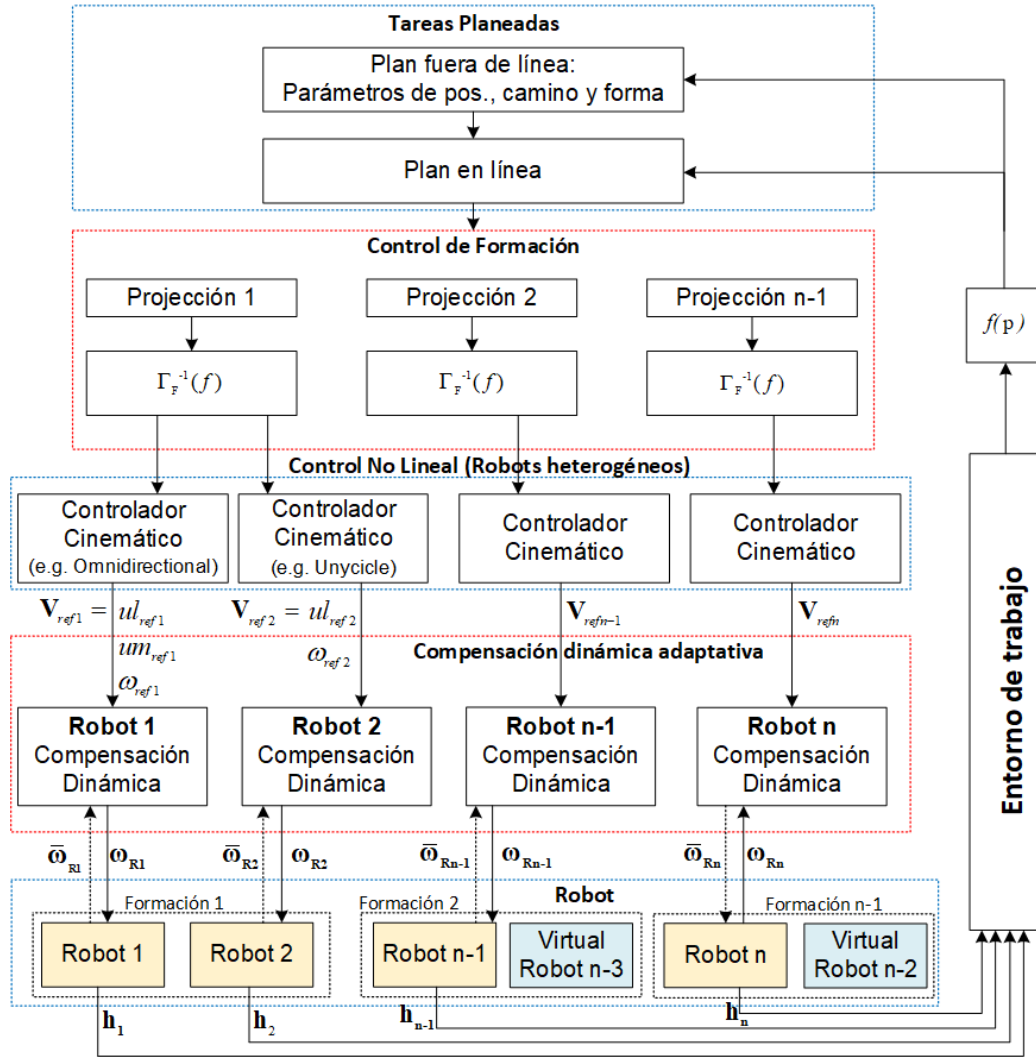


Figura 6.23. Sistema de control Multicapas.

6.2.2 Transformación Cinemática.

Dado que las formaciones están compuestas por proyecciones entre un par de robots, es necesaria una formación adicional para incluir un tercer robot, lo que requiere el uso de un robot virtual para generar la proyección adicional. Se supone que el objeto que se va a transportar o la forma que debe tener el conjunto robótico no tiene una geometría definida, puede tener una distribución de peso desigual o puede estar moviéndose en un espacio semiestructurado. En este aspecto, el cumplimiento de tareas con este tipo de objetivo puede presentar mejores resultados si los mecanismos robóticos tienen características heterogéneas, como robots similares a los coches, omnidireccionales o diferenciales. La ubicación de cada uno de los robots depende totalmente de los requerimientos de las tareas, pudiendo alterar las configuraciones en cualquier momento de tiempo si el espacio de trabajo lo requiere.

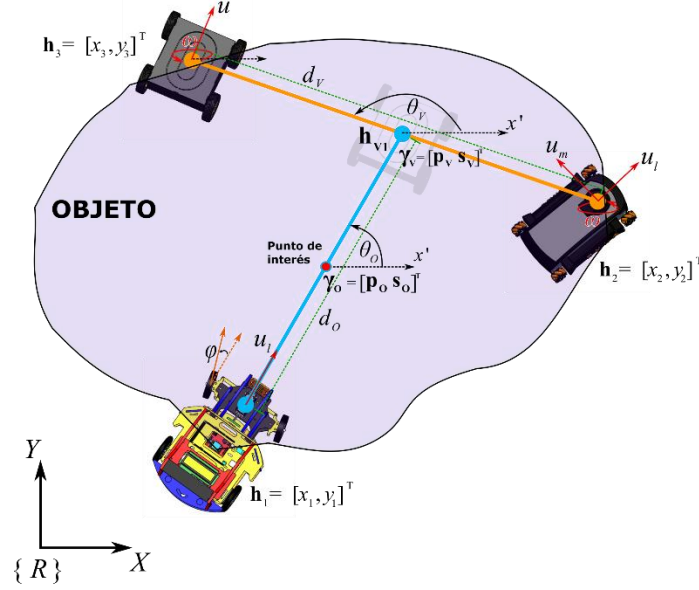


Figura 6.24. Análisis de formación entre robots heterogéneos.

Basándose en la formación presentada para el análisis (Figura 6.24), la principal proyección para transportar un objeto por tres vehículos móviles se describe como $\gamma_o = [p_o \ s_o]^T \in \mathbb{R}^4$, donde $p_o = [x_o \ y_o]^T \in \mathbb{R}^2$ representa la posición del principal punto intermedio y $s_o = [\theta_o \ d_o]^T \in \mathbb{R}^2$ indica la forma de la proyección, ambas con respecto al eje de referencia $\{R\}$. Por lo tanto, p_o se encuentra entre la posición del primer robot físico h_1 y el primer robot virtual h_{v1} . Por otra parte, para la segunda proyección $\gamma_v = [p_v \ s_v]^T \in \mathbb{R}^4$, el punto $p_v = [x_v \ y_v]^T \in \mathbb{R}^2$ se encuentra entre la posición del segundo h_2 y el tercer robot físico h_3 (el punto que también se representa como el robot virtual h_{v1}), mientras que $s_v = [d_v \ \theta_v]^T$ describe la forma secundaria. Las características del objeto transportado están definidas por las distancias y los ángulos deseados que se forman entre cada uno de los robots, definidos por $s_o = [d_o \ \theta_o]^T$ y $s_v = [d_v \ \theta_v]^T$, donde d_o representa la separación entre el primer robot y la posición virtual, d_v representa la separación entre los otros dos robots físicos h_2 y h_3 , mientras que θ_o y θ_v representan la orientación de ambas proyecciones con respecto al eje Y y al eje X desde el plano de referencia $\{R\}$.

La relación entre la formación pose-forma de cada proyección y las posiciones de punto de interés de cada uno de los vehículos viene dada por la transformación cinemática directa e inversa, es decir, $\gamma_k = f(g_k)$ y $g_k = f^{-1}(\gamma_k)$, donde $\gamma_k = [p_k \ s_k]^T$ y $g_k = [h_{1st}^T \ h_{2nd}^T]^T$, con $k = 0, V$ (formaciones principal y secundaria, respectivamente) para este caso, mientras que h_{1st}^T y h_{2nd}^T son el primer y segundo robot de la proyección, respectivamente. Las transformaciones cinemáticas directas $f(\cdot)$ para ambas proyecciones están dadas por:

$$\boldsymbol{\gamma}_V = \begin{bmatrix} \mathbf{p}_V \\ \mathbf{s}_V \end{bmatrix}; \mathbf{p}_V = \begin{bmatrix} x_V \\ y_V \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(x_3 + x_2) \\ \frac{1}{2}(y_3 + y_2) \end{bmatrix}; \mathbf{s}_V = \begin{bmatrix} d_V \\ \theta_V \end{bmatrix} = \begin{bmatrix} \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} \\ \tan^{-1} \left(\frac{y_3 - y_2}{x_3 - x_2} \right) \end{bmatrix}, \quad (6.14)$$

$$\boldsymbol{\gamma}_O = \begin{bmatrix} \mathbf{p}_O \\ \mathbf{s}_O \end{bmatrix}; \mathbf{p}_O = \begin{bmatrix} x_O \\ y_O \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(x_V + x_1) \\ \frac{1}{2}(y_V + y_1) \end{bmatrix}; \mathbf{s}_O = \begin{bmatrix} d_O \\ \theta_O \end{bmatrix} = \begin{bmatrix} \sqrt{(x_V - x_1)^2 + (y_V - y_1)^2} \\ \tan^{-1} \left(\frac{y_V - y_1}{x_V - x_1} \right) \end{bmatrix}. \quad (6.15)$$

A su vez, la transformación cinemática inversa $f(\cdot)$ es denotada por:

$$\mathbf{g}_V = \begin{bmatrix} \mathbf{h}_3 \\ \mathbf{h}_2 \end{bmatrix} = \begin{bmatrix} x_V + \frac{1}{2}d_V \cos \theta_V \\ y_V + \frac{1}{2}d_V \sin \theta_V \\ x_V - \frac{1}{2}d_V \cos \theta_V \\ y_V - \frac{1}{2}d_V \sin \theta_V \end{bmatrix} \text{ and} \quad (6.16)$$

$$\mathbf{g}_O = \begin{bmatrix} \mathbf{p}_V \\ \mathbf{h}_1 \end{bmatrix} = \begin{bmatrix} x_O + \frac{1}{2}d_O \cos \theta_O \\ y_O + \frac{1}{2}d_O \sin \theta_O \\ x_O - \frac{1}{2}d_O \cos \theta_O \\ y_O - \frac{1}{2}d_O \sin \theta_O \end{bmatrix} \quad (6.17)$$

Observación 1: x_O, y_O, x_V, y_V se encuentran en el marco de la referencia inercial.

La relación entre la variación de $\mathbf{g}(t)$ y $\boldsymbol{\gamma}(t)$ puede obtenerse a través de la derivada temporal de las transformaciones cinemáticas hacia adelante y hacia atrás, representadas por la matriz Jacobiana $\boldsymbol{\Gamma}_F(\mathbf{g})$ definida como:

$$\dot{\boldsymbol{\gamma}}(t) = \boldsymbol{\Gamma}_F(\mathbf{g})\dot{\mathbf{g}}(t) \quad (6.18)$$

y en la forma inversa está dada por:

$$\dot{\mathbf{g}}(t) = \boldsymbol{\Gamma}_F^{-1}(\boldsymbol{\gamma})\dot{\boldsymbol{\gamma}}(t), \quad (6.19)$$

donde:

$$\boldsymbol{\Gamma}_F(\mathbf{g}) = \frac{\partial \boldsymbol{\gamma}_{fx1}}{\partial \mathbf{g}_{ex1}} \text{ y } \boldsymbol{\Gamma}_F^{-1}(\boldsymbol{\gamma}) = \frac{\partial \mathbf{g}_{ex1}}{\partial \boldsymbol{\gamma}_{fx1}}, \text{ con } e, f = 4.$$

6.2.3 Diseño de los controladores.

a. Análisis del controlador considerando los objetivos secundarios.

El diseño de los controladores cinemáticos propuestos en este documento se basó en herramientas de métodos numéricos. Para facilitar la búsqueda de la solución de un conjunto de ecuaciones, un sistema puede ser representado en una estructura matricial, donde se aplican los teoremas y axiomas del álgebra lineal. De esta manera, se considera la ecuación diferencial de primer orden:

$$\dot{\eta}(t) = f(\eta(t), \zeta(t)), \quad (6.20)$$

donde $\eta(t)$ representa la salida del sistema para ser el controlador con condiciones iniciales $\eta(0) = \eta_0$ es la primera derivada con respecto al tiempo, y $\zeta(t)$ es la acción de control. Además, $\eta(t)$ se convierte en $\eta(k)$ en el tiempo discreto con $t = kT_0$, donde T_0 representa el tiempo de muestreo propuesto respetando el teorema de Nyquist, y k son las muestras de la respuesta continua.

Dado que se conocen el estado y la acción de control en el instante de tiempo $t(k-1)$, el estado del sistema en el instante $t(k)$ puede ser aproximado por el método de Euler como:

$$\frac{\eta(k) - \eta(k-1)}{T_0} = f(\eta(t), \zeta(t)), \quad (6.21)$$

El diseño del controlador cinemático se basó en la transformación cinemática de la cooperación de los vehículos móviles. Para diseñar un controlador de formación o un camino a seguir, la transformación cinemática puede ser aproximada como:

$$\frac{\eta(k) - \eta(k-1)}{T_0} = \Gamma(\zeta(k))\zeta(k) \quad (6.22)$$

donde $\Gamma(\zeta(k))$ contiene las características tanto del posicionamiento como de la formación de todos los sistemas robóticos en el caso del controlador de formación o la información de posicionamiento del objeto en el caso de la trayectoria que sigue.

Es importante mencionar que seguir un camino consiste en mantener la posición y orientación del objeto a transportar dentro de una ruta predefinida sin parametrización en el tiempo. De esta manera, el objetivo del control es posicionar el objeto en el punto más cercano del camino marcado \mathcal{P} a la velocidad deseada v_d . Para alcanzar este objetivo, se considera la siguiente expresión,

$$\frac{\eta(k) - \eta(k-1)}{T_0} = v_d + \frac{W(\eta_d(k-1) - \eta(k-1))}{T_0} \quad (6.23)$$

donde $\eta_d(k-1)$ es un punto del camino deseado con la forma deseada en el instante anterior de k y $W(\tilde{\eta}(k-1))$ es una matriz diagonal que pondera los errores de control $\tilde{\eta}(k-1) = \eta_d(k-1) - \eta(k-1)$, definido como:

$$W(\tilde{\eta}_m(k-1)) = \frac{w_m}{1+|\tilde{\eta}_m(k-1)|'} \quad (6.24)$$

con $\tilde{\eta}$ representa el vector de error en las salidas del controlador (véase la Figura 6.25), y m que representa cada uno de los estados de formación del control cooperativo.

Ahora, se puede considerar que las ecuaciones (6.22) y (6.23) generan el sistema de ecuaciones:

$$\Gamma(\zeta(k))\dot{\zeta}(k) = \mathbf{v}_d + \frac{W(\eta_d(k-1) - \eta(k-1))}{T_0}, \quad (6.25)$$

que nos permite reescribir el sistema como $\mathbf{A}\mathbf{u} = \mathbf{b}$, donde $\mathbf{A} = \Gamma(\zeta(k)) \in \mathbb{R}^{m \times n}$, $\mathbf{u} = \dot{\zeta}(k) \in \mathbb{R}^m$, y $\mathbf{b} = \mathbf{v}_d + \frac{W(\eta_d(k-1) - \eta(k-1))}{T_0}$, con $\mathbf{b} \in \mathbb{R}^m$. De esta manera, las acciones de control se definen como $\mathbf{u} = \mathbf{A}^{-1}\mathbf{b}$. En el caso de que \mathbf{A} sea cuadrática ($\det(\mathbf{A}) \neq 0$), \mathbf{A} tiene una solución directa inversa, de lo contrario, debe ser necesario utilizar un método para resolver el problema pseudoinverso, dado por:

$$\underbrace{\dot{\zeta}(k)}_{\mathbf{u}} = \underbrace{\Gamma(\zeta(k))^T (\Gamma(\zeta(k))\Gamma(\zeta(k))^T)^{-1}}_{\mathbf{A}^{-1}} \underbrace{\left(\dot{\eta}_d(k) + \frac{W(\eta_d(k-1) - \eta(k-1))}{T_0} \right)}_{\mathbf{b}}. \quad (6.26)$$

Observación 2: El pseudo-inverso aquí aplicado fue el introducido por Moore y Penrose. Tiene la propiedad de que distribuye así los coeficientes de las columnas redundantes en la solución de que la suma de los cuadrados de estos coeficientes se minimizó. Desgraciadamente, en los casos en que pierde rango, se sabe que el enfoque pseudo-inverso no siempre evitará las configuraciones singulares. En las singularidades, la tarea original es reemplazada por una solución que no corresponde precisamente a la original. Mediante el uso del algoritmo Gram-Schmidt o de la Descomposición del Valor Singular, se pueden construir pseudo-inversos en los que se pueden evitar las singularidades, pero eso está fuera del alcance de este trabajo.

Además, se dice que un sistema de ecuaciones lineales es homogéneo si puede escribirse en la forma $\mathbf{A}\mathbf{u} = \mathbf{0}$. Ahora bien, suponiendo que la configuración del sistema robótico (6.26) es redundante, la matriz jacobiana $\Gamma(\zeta(k)) \in \mathbb{R}^{m \times n}$ tiene más incógnitas que las ecuaciones ($m < n$) con rango $r = n$ para cada \mathbf{b} , y teniendo en cuenta que la ecuación homogénea tiene una solución no trivial, el sistema podría tener soluciones infinitas. En este caso, supongamos que la ecuación $\mathbf{A}\mathbf{u} = \mathbf{b}$ es consistente para un \mathbf{b} dado y dejando que \mathbf{v}_p sea una solución particular, la solución es el conjunto de todos los vectores de la forma

$$\mathbf{v} = \mathbf{v}_p + \mathbf{v}_h, \quad (6.27)$$

con \mathbf{v}_h como cualquier solución del sistema homogéneo $\Gamma(\zeta(k))\mathbf{v}_h = \mathbf{0}$.

Un método de solución viable es formular el problema como un problema de optimización lineal restringida

$$\frac{1}{2} \|\mathbf{v}\|_2^2 = \min, \quad (6.28)$$

que da lugar a la solución particular

$$\mathbf{v}_p = \mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T (\mathbf{\Gamma}(\boldsymbol{\zeta}(k))\mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T)^{-1} \mathbf{b}. \quad (6.29)$$

por otro lado, el espacio nulo de $\mathbf{\Gamma}(\boldsymbol{\zeta}(k))$ en \mathbb{R}^n es el conjunto de velocidades que no producen ningún efecto sobre las acciones del heterogéneo conjunto robótico. En ese caso, es necesario repensar la función de coste, expresándola como

$$\frac{1}{2} \|\mathbf{v} - \mathbf{v}_0\|_2^2 = \min. \quad (6.30)$$

que produce la solución homogénea

$$\mathbf{v}_h = \left(\mathbf{I}_n - \mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T (\mathbf{\Gamma}(\boldsymbol{\zeta}(k))\mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T)^{-1} \mathbf{\Gamma}(\boldsymbol{\zeta}(k)) \right) \mathbf{v}_0. \quad (6.31)$$

De este modo, insertando las ecuaciones (6.29) y (6.31) en (6.27), se obtiene la ley de control propuesta

$$\mathbf{v}_c = \underbrace{\mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T (\mathbf{\Gamma}(\boldsymbol{\zeta}(k))\mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T)^{-1} \mathbf{b}}_{\mathbf{v}_p} + \underbrace{\left(\mathbf{I}_n - \mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T (\mathbf{\Gamma}(\boldsymbol{\zeta}(k))\mathbf{\Gamma}(\boldsymbol{\zeta}(k))^T)^{-1} \mathbf{\Gamma}(\boldsymbol{\zeta}(k)) \right) \mathbf{v}_0}_{\mathbf{v}_h}, \quad (6.32)$$

donde el primer término a la izquierda es la solución particular \mathbf{v}_p y el segundo término \mathbf{v}_h de esta ecuación pertenecen al espacio nulo de $\mathbf{\Gamma}(\boldsymbol{\zeta}(k))$.

Para los propósitos de este trabajo, el segundo término de la ecuación (6.32) representa la proyección sobre el espacio nulo de los sistemas robóticos, donde \mathbf{v}_0 es un vector arbitrario que contiene las velocidades asociadas con la forma y las orientaciones del conjunto robótico. Por lo tanto, cualquier valor dado a \mathbf{v}_0 afectará sólo a la estructura interna de la formación, pero no al control final del primer objetivo en absoluto. El espacio nulo creado por la matriz de tareas de alto nivel permite que cada velocidad se proyecte en ese espacio, donde las subtareas compiten para resolver el problema de diferentes maneras. Sin embargo, las velocidades de la segunda tarea deben ser calculadas e incluidas en \mathbf{v}_0 , siendo

$$\mathbf{v}_0 = \mathbf{\Gamma}_s(\boldsymbol{\zeta}(k)) \left(\mathbf{\Gamma}_s(\boldsymbol{\zeta}(k))\mathbf{\Gamma}_s(\boldsymbol{\zeta}(k))^T \right)^{-1} \left(\dot{\boldsymbol{\eta}}_d(k) + \frac{\mathbf{w}_s(\boldsymbol{\eta}_{ds}(k-1) - \boldsymbol{\eta}_s(k-1))}{T_0} \right), \quad (6.33)$$

donde $\mathbf{\Gamma}_s(\boldsymbol{\zeta}(k))$ es la matriz jacobiana, que contiene los objetivos secundarios.

Observación 3: La consideración de tener más de un robot trabajando en el mismo propósito permite lograr tareas secundarias al mismo tiempo colectivamente. Para lograr estos objetivos, se puede implementar un controlador basado en el comportamiento para dividir las tareas en subobjetivos, resolviendo los problemas por separado y finalmente combinándolos para obtener la solución final.

6.2.4 Control de Formación y seguimiento de caminos.

El control cooperativo de robots móviles heterogéneos considera que el tipo de robots incorporados al sistema de formación incluye su propio controlador cinemático y un compensador dinámico, por lo tanto, para el control de la formación, el tipo de sistema incluido es transparente. En esta sección se describe brevemente el esquema básico de control para mantener la formación de muchos sistemas robóticos, base para integrar más robots de manera escalable. La Figura 6.25 muestra la estructura de control implementada para este trabajo, denotando tanto los controladores que mantienen la formación, orientación y seguimiento del camino deseado como los controladores que corrigen las acciones finales de cada robot.

La adopción del enfoque del espacio nulo se debió a la posibilidad de tratar las tareas y subtareas por separado, que al final se unificaron para obtener las acciones de control. Dado que la tarea principal no entra en conflicto con la tarea secundaria, para el diseño de los controladores se seleccionó la jerarquía de tareas. De esta manera se pueden conseguir diferentes objetivos de control, por ejemplo, máxima manipulabilidad, ahorro de energía, evitar obstáculos, etc. En este caso, la tarea principal era lograr que el conjunto de robots siguiera un camino, mientras que la tarea secundaria era mantener la forma geométrica y la orientación de cada proyección formada por los vehículos.

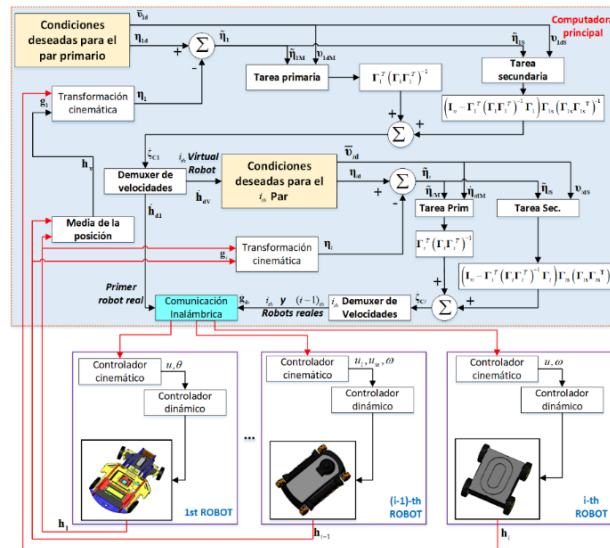


Figura 6.25. Estructura de control implementada.

La matriz jacobina unificada de la ecuación (6.22) contiene tanto las primeras derivadas correspondientes a las posiciones del punto central de una formación, como la distancia y la orientación entre los dos robots que forman la proyección. Para el cálculo de las diferentes acciones de control, se tiene en cuenta que la matriz jacobiana de la ecuación (6.22) se divide en dos partes:

$$\begin{array}{c} \dot{\boldsymbol{\eta}}_{1M} \\ \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{d}_o \\ \dot{\theta}_o \end{bmatrix} \\ \dot{\boldsymbol{\eta}}_{1S} \end{array} = \begin{array}{c} \boldsymbol{\Gamma}_{1M} \\ \begin{bmatrix} \frac{\partial x_o}{\partial x_1} & \frac{\partial x_o}{\partial y_1} & \frac{\partial x_o}{\partial x_{v1}} & \frac{\partial x_o}{\partial y_{v1}} \\ \frac{\partial y_o}{\partial x_1} & \frac{\partial y_o}{\partial y_1} & \frac{\partial y_o}{\partial x_{v1}} & \frac{\partial y_o}{\partial y_{v1}} \\ \frac{\partial d_o}{\partial x_1} & \frac{\partial d_o}{\partial y_1} & \frac{\partial d_o}{\partial x_{v1}} & \frac{\partial d_o}{\partial y_{v1}} \\ \frac{\partial \theta_o}{\partial x_1} & \frac{\partial \theta_o}{\partial y_1} & \frac{\partial \theta_o}{\partial x_{v1}} & \frac{\partial \theta_o}{\partial y_{v1}} \end{bmatrix} \\ \boldsymbol{\Gamma}_{1S} \end{array} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_{v1} \\ \dot{y}_{v1} \end{bmatrix} \quad (6.34)$$

donde $\boldsymbol{\Gamma}_{1M}(\boldsymbol{\zeta}(k)) \in R^{2 \times 4}$ son las primeras derivadas de las posiciones del punto medio de la proyección, $\boldsymbol{\Gamma}_{1S}(\boldsymbol{\zeta}(k)) \in R^{2 \times 4}$ son las primeras derivadas de la distancia y el ángulo de orientación, $\dot{\boldsymbol{\eta}}_{1M} \in R^{2 \times 1}$ corresponde a la primera variable a calcular y $\dot{\boldsymbol{\eta}}_{1S} \in R^{2 \times 1}$ la segunda variable a calcular.

Teniendo en cuenta la división propuesta en la ecuación (6.34), se pueden proponer controles diferenciados para lograr tanto los objetivos primarios como los secundarios. El trabajo propone como objetivo principal del punto de interés \mathbf{p}_o (redefinido como $\boldsymbol{\eta}_{1M}$ para la aplicación del controlador propuesto) sigue un perfil no parametrizado en el tiempo, con una velocidad \mathbf{v}_d impuesta por el diseñador. Por otro lado, el objetivo secundario era modificar la formación de los sistemas robóticos, siempre y cuando no entraran en conflicto con la satisfacción del primer objetivo. Teniendo una formación basada en proyecciones, la proyección principal $\boldsymbol{\gamma}_o$ juega un papel importante ya que es la base para el cálculo de las velocidades que el resto de las proyecciones adoptarán para cumplir las tareas. Además, tener una línea recta principal que puede ser proyectada y orientada según los requerimientos del diseñador del controlador permite localizar los sistemas robóticos que se acuerden. Contrasta las posibilidades a) y b) de la Figura 6.26; la primera proyección tiene una longitud y orientación diferente a la segunda proyección. Sin embargo, el objetivo principal se logró en ambas, donde la variación de las distancias y las orientaciones secundarias no entraron en conflicto con la tarea principal. En los párrafos siguientes se describen con más detalle las características mencionadas.

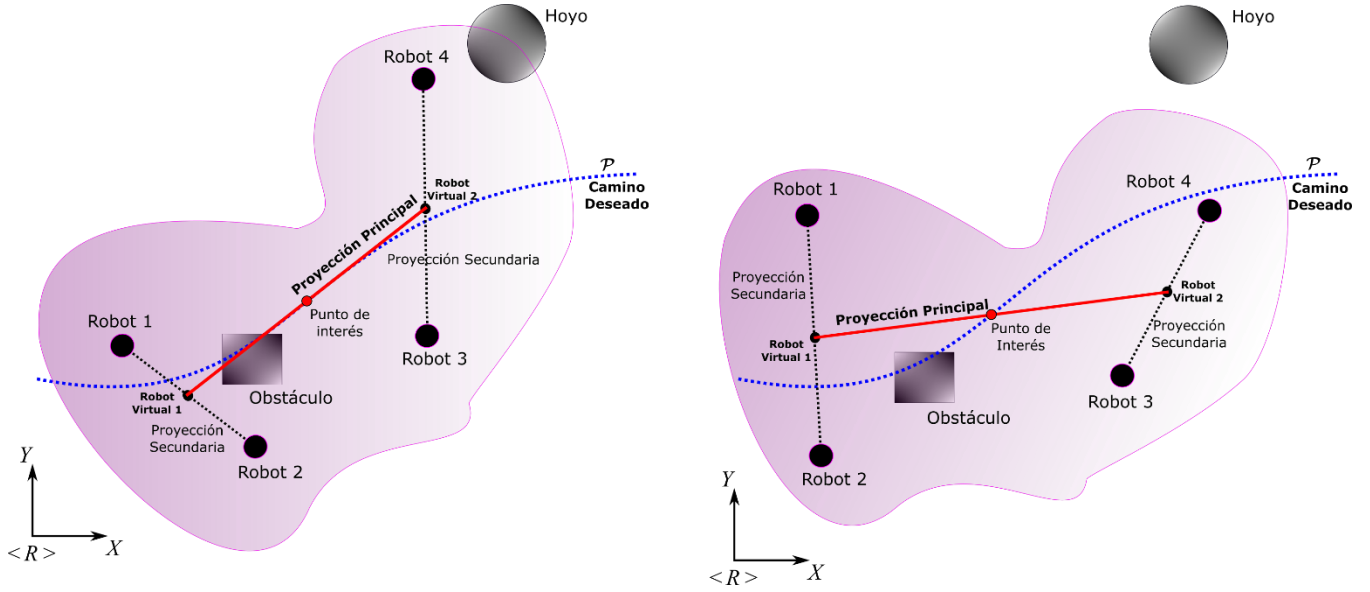


Figura 6.26 Posibilidades de control considerando el seguimiento del camino y la formación del conjunto robótico.

a. Controlador de seguimiento del camino.

Para lograr que el centro de la proyección η_{iM} siga el camino deseado, el controlador propuesto está definido por

$$\dot{\zeta}_{iM}(k) = \Gamma_{iM}(\zeta(k))^T \left(\Gamma_{iM}(\zeta(k)) \Gamma_{iM}(\zeta(k))^T \right)^{-1} \left(\bar{v}_{idM} + W_{iM}(\tilde{\eta}_{iM}(k-1)) \right), \quad (6.35)$$

donde $\Gamma_{iM}(\zeta(k))^T$ encuentra la primera parte del jacobiano general que determina las características del conjunto robótico para seguir el camino deseado, con $\tilde{\eta}_{iM}(k-1) = \eta_{diM}(k-1) - \eta_{iM}(k-1)$. La respuesta obtenida en $\dot{\zeta}_{iM}(k)$ son dos pares de velocidades que se aplicarán a los robots situados en los extremos de la proyección, ya sean físicos o virtuales. Estas velocidades se deben a que mantienen el punto de interés en el camino deseado, pero dejando de lado los errores de la distancia entre los robots y el ángulo formado con respecto a $\langle R \rangle$. Es importante señalar que la proyección principal (donde se encuentra el punto de interés del objeto transportado) puede tener la velocidad deseada, en este caso, $\bar{v}_{idM} = v_{dM}$, mientras que el resto de proyecciones tendrán velocidades generadas por el controlador principal, es decir, $\bar{v}_{idM} = \dot{\eta}_{idM}$.

b. Control de la formación y la orientación.

Esta parte del control permite establecer la formación deseada η_{iS} como objetivo secundario, proporcionando flexibilidad a las modificaciones de proyección entre los robots en aplicaciones donde las tareas de los objetivos de rastreo eran necesarias, por ejemplo, dada la presencia de los obstáculos en la carretera. La segunda parte de la matriz jacobiana se considera en este

controlador, que proporciona las características de formación y orientación $\mathbf{r}_{is}(\zeta(k))$ de ambos sistemas robóticos. De esta manera, el controlador se define por

$$\dot{\zeta}_2(k) = \mathbf{r}_{is}(\zeta(k))^T \left(\mathbf{r}_{is}(\zeta(k)) \mathbf{r}_{is}(\zeta(k))^T \right)^{-1} \left(v_{as} + \mathbf{W}_2(\tilde{\eta}_{is}(k-1)) \right). \quad (6.36)$$

Al contrario que a), este controlador tiene como objetivo corregir la forma de la proyección i_{th} , poniendo a cero los errores de distancia entre los robots, y la orientación general del par de robots.

c. Controlador Unificado.

Esta sección define la prioridad en las tareas de control que tendrá el grupo de robots. El trabajo propone el control del posicionamiento del punto de interés (en este caso, el punto medio de la proyección principal). Por lo tanto, la tarea que no entra en conflicto con el objetivo principal de control es la forma y la orientación de la proyección. Basándose en (6.32), el controlador final propuesto se presenta en la forma

$$\begin{aligned} \dot{\zeta}_c(k) = & \mathbf{r}_{iM}(\zeta)^T (\mathbf{r}_{iM}(\zeta) \mathbf{r}_{iM}(\zeta)^T)^{-1} \left(\bar{v}_{idM} + \frac{\mathbf{W}(\tilde{\eta}_{iM}(k-1))}{T_0} \right) + \\ & (\mathbf{I}_n - \mathbf{r}_{iM}(\zeta)^T (\mathbf{r}_{iM}(\zeta) \mathbf{r}_{iM}(\zeta)^T)^{-1} \mathbf{r}_{iM}(\zeta)) \mathbf{r}_{is}(\zeta) (\mathbf{r}_{is}(\zeta) \mathbf{r}_{is}(\zeta)^T)^{-1} \left(v_{ids} + \frac{\mathbf{W}_s(\tilde{\eta}_{is}(k-1))}{T_0} \right), \end{aligned} \quad (6.37)$$

donde la segunda parte de este controlador muestra el intento de corregir los errores siempre y cuando no interfiera con la tarea principal, que es llevada a cabo por la primera parte del controlador. Esta fusión permite tener las velocidades deseadas para el par de robots que forman la proyección i_{th} .

Observación 4: Los casos a), b) y c) son aplicables a una de las i posibles proyecciones, lo que requiere que los controladores i sean programados como proyecciones existentes. Esto facilita la escalabilidad del sistema, al tiempo que hace más flexible el conjunto de los robots al dar los parámetros deseados a cada una de las proyecciones en cualquier momento.

Observación 5: En el estudio de caso, v_{ids} puede ser impuesto por el diseñador del controlador, pero debe estar directamente relacionado con la reducción del error, es decir, se puede configurar una velocidad con la que se corrige la forma a los parámetros deseados, pero debe tener un valor nulo en caso de que se logren los objetivos, de lo contrario, pueden aparecer errores causados por la misma acción de corrección. Una forma de resolver esto es forzarlo a que se sumerja dentro del peso dado por \mathbf{W}_s .

d. Análisis de estabilidad.

Reescribiendo la propuesta de ley de control (6.37) de la siguiente manera

$$\dot{\zeta}_c(k) = \Gamma_M^\# \left(\bar{v}_{dM} + \frac{W_M(\tilde{\eta}_M(k-1))}{T_0} \right) + (I - \Gamma_M^\# \Gamma_M) \Gamma_S^\# \left(v_{dS} + \frac{W_S(\tilde{\eta}_S(k-1))}{T_0} \right), \quad (6.38)$$

donde $\Gamma_M^\# = \Gamma_M(\zeta)^T (\Gamma_M(\zeta) \Gamma_M(\zeta)^T)^{-1}$ y $\Gamma_S^\# = \Gamma_S(\zeta)^T (\Gamma_S(\zeta) \Gamma_S(\zeta)^T)^{-1}$ representan una notación de matriz pseudo-inversa, el análisis puede ser desarrollado de una manera más simple.

Ahora bien, el análisis se hace para el control del objetivo principal, además, se considera que el objetivo secundario de la formación no se ve afectado cuando se multiplica la Ecuación (6.38) por Γ_M :

$$\Gamma_M \dot{\zeta}_c(k) = \Gamma_M \Gamma_M^\# \left(\bar{v}_{dM} + \frac{W_M(\tilde{\eta}_M(k-1))}{T_0} \right) + \Gamma_M (I - \Gamma_M^\# \Gamma_M) \Gamma_S^\# \left(v_{dS} + \frac{W_S(\tilde{\eta}_S(k-1))}{T_0} \right) \quad (6.39)$$

Entonces, como $\Gamma_M \Gamma_M^\# = I$ y $\Gamma_M (I - \Gamma_M^\# \Gamma_M) = 0$, la ecuación (6.39) se reduce a:

$$\Gamma_M \dot{\zeta}_c(k) = \left(\bar{v}_{dM} + \frac{W_M(\tilde{\eta}_M(k-1))}{T_0} \right); \quad (6.40)$$

El análisis de estabilidad para la Ecuación (6.40) se realiza mediante la retroalimentación, es decir, si se considera que la Ecuación (6.34) es de seguimiento perfecto entonces, $\dot{\eta}_M(k) \cong \Gamma_M \dot{\zeta}_c(k)$, y teniendo en cuenta que el error para el objetivo primario se define como $\eta_{dM}(k-1) - \eta_M(k-1)$ tal que

$$\dot{\eta}_M(k) = \bar{v}_{dM} + W_M \left(\frac{\eta_{dM}(k-1) - \eta_M(k-1)}{T_0} \right); \quad (6.41)$$

por lo tanto, sustituyendo la velocidad para el objetivo principal $\dot{\eta}_M(k) = \frac{\Delta \dot{\eta}_M(k)}{T_0} = \frac{\eta_M(k) - \eta_M(k-1)}{T_0}$ en la Ecuación (6.41) y también considerando que la velocidad deseada para el camino es $\bar{v}_{dM} = \dot{\eta}_{dM} - \dot{\gamma}_M$:

$$\frac{\eta_M(k) - \eta_M(k-1)}{T_0} = \dot{\eta}_{dM} - \dot{\gamma}_M + W_M \left(\frac{\eta_{dM}(k-1) - \eta_M(k-1)}{T_0} \right). \quad (6.42)$$

A continuación, se aplica la derivada diferencial para obtener $\dot{\eta}_{dM}(k) = \frac{\eta_{dM}(k) - \eta_{dM}(k-1)}{T_0}$ y de la misma manera a $\dot{\gamma}_M(k) = \frac{\Delta \gamma_M}{T_0}$, luego la expresión

$$\frac{\eta_M(k) - \eta_M(k-1)}{T_0} = \frac{\eta_{dM}(k) - \eta_{dM}(k-1)}{T_0} - \frac{\Delta \gamma_M}{T_0} + W_M \left(\frac{\eta_{dM}(k-1) - \eta_M(k-1)}{T_0} \right) \quad (6.43)$$

Simplificando T_0 y agrupando la Ecuación (6.43) con los estados actuales (k) y el estado previo ($k - 1$) hay

$$(\boldsymbol{\eta}_{dM}(k-1) - \boldsymbol{\eta}_M(k-1)) = (\boldsymbol{\eta}_{dM}(k) - \boldsymbol{\eta}_M(k)) - \Delta\boldsymbol{\gamma}_M + \mathbf{W}_M(\boldsymbol{\eta}_{dM}(k-1) - \boldsymbol{\eta}_M(k-1)) \quad (6.44)$$

Ahora, teniendo en cuenta que $\boldsymbol{\eta}_{dM}(k) - \boldsymbol{\eta}_M(k) = \tilde{\boldsymbol{\eta}}_M(k)$ y $\boldsymbol{\eta}_{dM}(k-1) - \boldsymbol{\eta}_M(k-1) = \tilde{\boldsymbol{\eta}}_M(k-1)$ representan el error actual y el error anterior, respectivamente, la Ecuación (6.44) se reduce a

$$\tilde{\boldsymbol{\eta}}_M(k-1) = \tilde{\boldsymbol{\eta}}_M(k) - \Delta\boldsymbol{\gamma}_M + \mathbf{W}_M(\tilde{\boldsymbol{\eta}}_M(k-1)), \quad (6.45)$$

y agrupando términos semejantes, resulta que

$$\Delta\boldsymbol{\gamma}_M = \tilde{\boldsymbol{\eta}}_M(k) + \mathbf{W}_M(\tilde{\boldsymbol{\eta}}_M(k-1)) - \tilde{\boldsymbol{\eta}}_M(k-1). \quad (6.46)$$

El error objetivo primario $\tilde{\boldsymbol{\eta}}_M(k)$ se separa para tener la siguiente expresión

$$\Delta\boldsymbol{\gamma}_M = \tilde{\boldsymbol{\eta}}_M(k) + \tilde{\boldsymbol{\eta}}_M(k-1)(\mathbf{W}_M - \mathbf{I}), \quad (6.47)$$

por lo tanto, la estabilidad en tiempo discreto se realiza por la transformada \mathcal{Z} , y teniendo en cuenta el análisis para cada valor de la matriz diagonal \mathbf{W}_M así

$$\begin{aligned} (1 - z^{-1})\boldsymbol{\gamma}_M(z) &= \tilde{\boldsymbol{\eta}}_M(z) + \tilde{\boldsymbol{\eta}}_M(z)z^{-1}(\mathbf{W}_M - \mathbf{I}) \\ \tilde{\boldsymbol{\eta}}(z) &= \frac{1-z^{-1}}{1+z^{-1}(\mathbf{W}_M-\mathbf{I})}\boldsymbol{\gamma}(z). \end{aligned} \quad (6.48)$$

Finalmente, se analizan los polos de la función obtenidos de la siguiente manera:

$$\begin{aligned} 1 + z^{-1}(\mathbf{W}_M - \mathbf{I}) &= 0 \\ z &= 1 - \mathbf{W}_M \end{aligned} \quad (6.49)$$

Por lo tanto, para que el control sea estable, el polo en el plano \mathcal{Z} debe estar dentro del círculo unitario. Por lo tanto, los valores de la matriz diagonal deben ser considerados como $0 < \text{diag}(\mathbf{W}_M) < 1$, de ahí el error de control $\tilde{\boldsymbol{\eta}}_M(k)$ se acercan a ser cero cuando $k \rightarrow \infty$, entonces el sistema es asintóticamente estable.

Ahora, como segundo paso, procedemos a efectuar el análisis para el objetivo secundario de la formación. De manera similar al objetivo primario, multiplicar ambos miembros de la ley de control por $\boldsymbol{\Gamma}_S$, luego la Ecuación (6.38) se escribe como:

$$\boldsymbol{\Gamma}_S \dot{\boldsymbol{\zeta}}_c(k) = \boldsymbol{\Gamma}_S \boldsymbol{\Gamma}_M^\# \left(\bar{\mathbf{v}}_{dM} + \frac{\mathbf{W}_M(\tilde{\boldsymbol{\eta}}_M(k-1))}{T_0} \right) + \boldsymbol{\Gamma}_S (\mathbf{I} - \boldsymbol{\Gamma}_M^\# \boldsymbol{\Gamma}_M) \boldsymbol{\Gamma}_S^\# \left(\mathbf{v}_{dS} + \frac{\mathbf{W}_S(\tilde{\boldsymbol{\eta}}_S(k-1))}{T_0} \right). \quad (6.50)$$

Cuando se analiza una propiedad importante de las matrices jacobinas cuando no existe conflicto entre ellas, es decir, que es factible cumplir las dos tareas de forma simultánea y completa, resulta que

$$\Gamma_S \Gamma_M^\# = 0. \quad (6.51)$$

Por lo tanto, si se sustituyen las ecuaciones (6.49) por (6.50), se obtiene la siguiente expresión:

$$\Gamma_S \dot{\zeta}_c(k) = \Gamma_S \Gamma_S^\# \left(v_{ds} + \frac{W_S(\tilde{\eta}_S(k-1))}{T_0} \right). \quad (6.52)$$

Además, considerando que la Ecuación (6.34) es un rastreo perfecto, está $\dot{\eta}_S(k) \cong \Gamma_S \dot{\zeta}_c(k)$. Realizando un procedimiento de análisis similar al de las Ecuaciones (6.40) en (6.52), se concluye de la misma manera que $z = I - W_S$, por lo tanto los valores de la matriz diagonal deben ser considerados como $0 < \text{diag}(W_S) < 1$, de ahí el error de control $\tilde{\eta}_S(k)$ se acercan a ser cero cuando $k \rightarrow \infty$, entonces el sistema es asintóticamente estable.

6.2.5 Control Cinemático y compensación dinámica en cada robot.

Las velocidades entregadas por el controlador de posición y formación son entradas para cada uno de los robots, donde los controladores locales interpretan independientemente las velocidades como el perfil $\dot{\mathbf{h}}_i$ a seguir. En esta sección se describe el control de cada uno de los heterogéneos robots que se podrían considerar en el conjunto, tanto para el control cinemático como para la corrección de los errores dados por la dinámica de cada sistema.

a. Control cinemático.

El modelo cinemático de un robot móvil da la ubicación de un punto de interés a cualquier parte del sistema robótico. El modelo cinemático instantáneo de un robot móvil da la derivada de su punto de interés en función de las derivadas de todo el sistema en sí.

$$\dot{\mathbf{h}}_i(t) = \mathbf{J}_i(\mathbf{q}) \mathbf{v}_i(t), \quad (6.53)$$

donde $\dot{\mathbf{h}}_i = [\dot{h}_{xi} \quad \dot{h}_{yi}]^T$ es el vector de las velocidades del punto de interés, $\mathbf{v}_i = [u_{li} \quad u_{mi} \quad \omega_{i...}]^T$ es el vector de las velocidades de los robots móviles, que contiene las velocidades lineales y angulares de la plataforma móvil, $\mathbf{J}_i(\mathbf{q})$ es la matriz jacobiana, que define un mapeo lineal entre el vector de las velocidades de los robots móviles $\mathbf{v}_i(t)$ y el vector de las velocidades de los puntos de interés, e i es el robot i_{th} del conjunto.

A través de la aproximación de Euler al modelo cinemático de cualquier robot móvil, se obtiene el siguiente modelo cinemático discreto:

$$\mathbf{h}(k+1) = \mathbf{h}(k) + T_0 \mathbf{J}(\mathbf{q}(k)) \mathbf{v}(k), \quad (6.54)$$

donde, los valores de \mathbf{h} en el tiempo discreto $t = kT_0$ se denotarán como $\mathbf{h}(k)$, con T_0 como tiempo de muestra, y $k \in \{0, 1, 2, 3, \dots\}$. A continuación, por la propiedad Markov y para ajustar el rendimiento de la ley de control propuesta, se sustituye el vector de estados $\mathbf{h}(k+1)$ por:

$$\mathbf{h}(k+1) = \mathbf{h}_d(k+1) - \mathbf{W}(\mathbf{h}_d(k) - \mathbf{h}(k)), \quad (6.55)$$

donde \mathbf{W} es una matriz diagonal, que debe satisfacer $0 < \text{diag}(W_{hx}, W_{hy}) < 1$, permitiendo reducir las variaciones de las variables de estado, y \mathbf{h}_d es la trayectoria deseada.

A continuación, a partir de las ecuaciones (6.54) y (6.55), se obtiene el siguiente sistema de ecuaciones lineales, que permite en cada instante de muestreo calcular las acciones de control

$$\mathbf{J}(\mathbf{q})\mathbf{v} = \mathbf{b}, \quad (6.56)$$

donde $\mathbf{v} = [u_l(k) \quad u_m(k) \quad \omega(k)]^T$ y:

$$\mathbf{b} = \frac{1}{T_0} \begin{bmatrix} h_{xd}(k+1) - w_{hx}(e_{hx}(k)) - h_x(k) \\ h_{yd}(k+1) - w_{hy}(e_{hy}(k)) - h_y(k) \end{bmatrix} \quad (6.57)$$

A partir de la ecuación (6.56), su solución por mínimos cuadrados se obtiene resolviendo las ecuaciones normales

$$\mathbf{v}_{ref} = \mathbf{J}(\mathbf{q})^T (\mathbf{J}(\mathbf{q})\mathbf{J}(\mathbf{q})^T)^{-1} \frac{1}{T_0} \begin{bmatrix} h_{xd}(k+1) - w_{hx}(e_{hx}(k)) - h_x(k) \\ h_{yd}(k+1) - w_{hy}(e_{hy}(k)) - h_y(k) \end{bmatrix} \quad (6.58)$$

b. Compensación dinámica.

Esta sección detalla cómo el controlador fue diseñado para compensar la dinámica del robot en el espacio de trabajo, logrando una mayor robustez para cada robot. Se programó un controlador dinámico dentro de cada plataforma móvil, es decir, las plataformas robóticas tienen un control interno que corrige los errores de velocidad lineal y angular.

Los errores lineales o angulares pueden ser causados por fuerzas o torques externos que afectan la dinámica de las plataformas. Dado que todos los robots recibieron los comandos de maniobra como entrada (lineales y angulares), el número de velocidades dependía de cada robot móvil para cumplir con los requisitos de formación. La Figura 6.27 describe cómo se estructura el controlador interno para compensar la dinámica de los robots. En este caso, el controlador genera una salida de control definida como:

$$\mu_{\omega i}(k) = K p_i \left\{ \tilde{\omega}_i(k) + K i_i T_o \sum_{j=1}^k \tilde{\omega}_i(j-1) + K_{D_i} \frac{\tilde{\omega}_i(k) - \tilde{\omega}_i(k-1)}{T_o} \right\}, \quad (6.59)$$

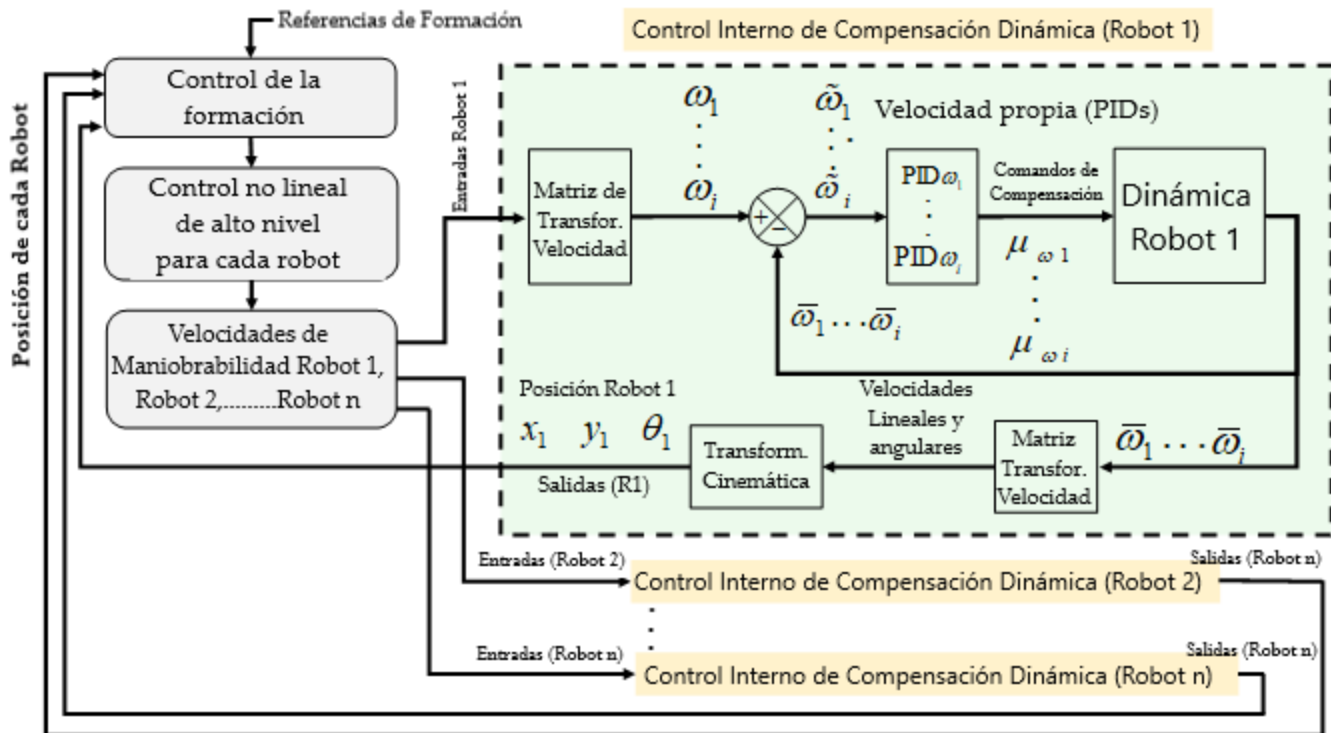


Figura 6.27 Esquema de la operación del control para compensación dinámica

En (6.59), $\tilde{\omega}_i(k) = \omega_i(k) - \bar{\omega}_i(k)$ representa el error de velocidad angular entre la velocidad angular deseada $\omega_i(k)$ y la velocidad angular real $\bar{\omega}_i(k)$ con $i = 1, 2, 3, \dots$ (i depende del número de ruedas de tracción controladas de cada robot), las variables Kp_i, Ki_i, KDi_i son las ganancias que pesan el error de control, y T_o es el tiempo de muestreo considerado para el PID discreto (control proporcional, integral y derivativo). Las matrices de transformación directa e inversa dependen de la configuración de cada robot móvil para lograr velocidades lineales, angulares y, eventualmente, la posición y orientación del robot móvil para retroalimentar el lazo de control del esquema cooperativo propuesto.

6.2.6 La instrumentación en los robots.

a. Sensores medidores de posición relativa.

Dentro de los múltiples sistemas robóticos, se utilizaron componentes electrónicos para alimentar las unidades de procesamiento incorporadas, conocer el estado de carga de las baterías, compartir información con receptores remotos de forma inalámbrica, conocer la odometría del sistema, etc. Algunos de estos sensores proporcionaban información sobre el estado de cada sistema robótico, permitiendo al robot situarse dentro de su espacio de trabajo para evitar colisiones. Los robots utilizados para los experimentos de este trabajo tienen sensores incorporados para estimar la posición dada una ubicación relativa, es decir, las ruedas de cada uno de los robots contienen codificadores que miden la velocidad y la dirección de rotación.

Estos sensores determinaron el desplazamiento angular, y por lo tanto el desplazamiento lineal de los robots, mostrando las limitaciones cuando hubo deslizamientos en tierra o la resolución de los codificadores no fue adecuada. El algoritmo de control propuesto considera el conocimiento de la posición actual de cada uno de los robots móviles en todo momento de la tarea; por lo tanto, al iniciar la tarea deseada, la condición inicial era la posición y orientación de cada robot que formaba parte de la tarea de colaboración.

b. Sistema de comunicaciones.

Como se explicó y demostró en el capítulo anterior, la estructura de comunicación propuesta para el conjunto de robots se basaba en el protocolo IEEE 802.15.4. Este protocolo permite la creación de redes multipunto, y también es capaz de transmitir una gran cantidad de datos (bytes) con una latencia muy baja y lograr una sincronización de comunicación predecible. Los módulos XBEE S2 permiten la aplicación del protocolo establecido, configurado de tal manera que todos ellos se encuentran dentro de la misma red inalámbrica. La topología utilizada en este trabajo fue la configuración en malla, es decir, todos los dispositivos pueden comunicarse entre sí de tal manera que cada robot tiene un identificador para saber qué paquete de datos le corresponde. Esto ofrece la ventaja de que, si en algún momento la comunicación falla o se pierde, la comunicación puede continuar entre todos los demás nodos (robots móviles) porque la conectividad se ve obligada a restaurarse. El coordinador de la red envía los comandos de maniobra a cada robot (velocidades lineales y angulares), donde las velocidades están empaquetadas dentro de una trama de datos. La Figura 6.28 muestra cómo el paquete de datos enviado por el robot maestro y el paquete que responde a los robots esclavos.

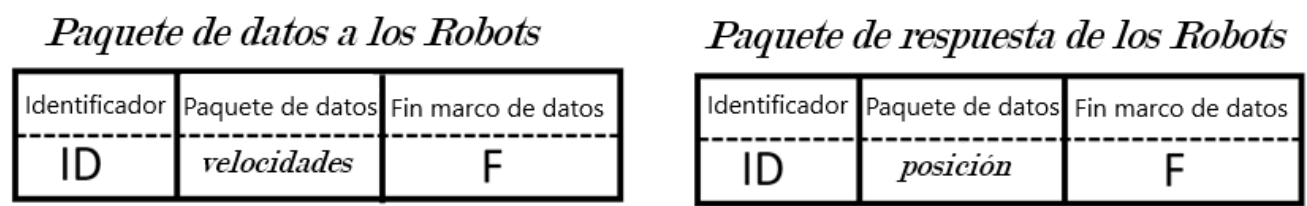


Figura 6.28. Trama de paquetes entre amos y esclavos de la red inalámbrica de comunicaciones

El esquema de comunicación propuesto cumple con una velocidad de muestreo de 175 ms, es decir, envía, recibe y procesa los datos de los robots dentro de la red propuesta. La Figura 6.29 muestra cómo se estructura la comunicación entre los robots en el sistema cooperativo. Las comunicaciones en este tipo de configuración MESH pueden eventualmente realizarse entre 65.000 dispositivos sin perder sus características, según los datos de los fabricantes. En este trabajo, el coste de aumentar los robots hace que las tramas sean más grandes, y eso implica una

reducción de la velocidad de las comunicaciones, que se establece no puede ser inferior a 100 ms. Aumentar el número de robots que superan este margen implicaría una aplicación de un uso masivo de robots, pero si este fuera el caso, habría que aumentar la restricción de las comunicaciones en la red, reduciendo la velocidad del propio sistema: por ejemplo, en una aplicación de transporte de objetos, el conjunto se movería a velocidades más bajas.

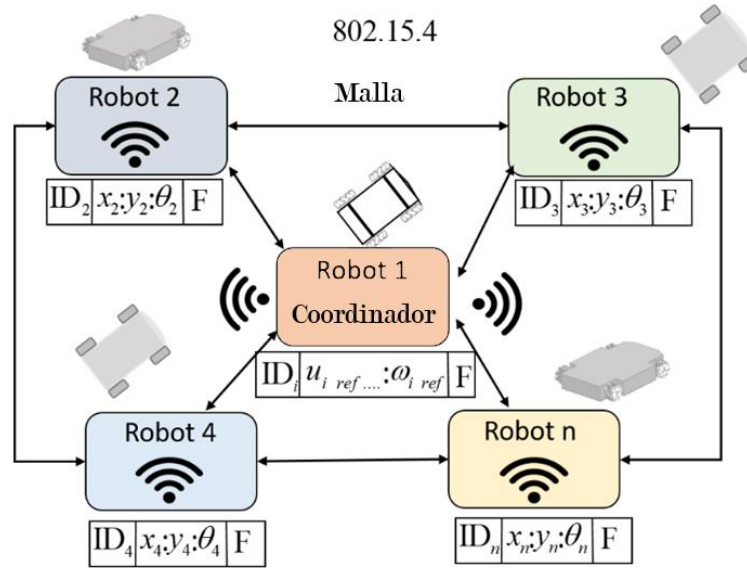


Figura 6.29. Estructura tipo malla usada en las comunicaciones

Observación 6: La codificación y el tamaño de los paquetes enviados a cada robot dependen del tipo de plataforma móvil, mientras que la información de retroalimentación tiene tamaños similares. Como en este trabajo se utilizó una configuración MESH, los robots pudieron intercambiar toda la información entre ellos, ya sea que fueran el destino o no; la recibieron para procesarla o sólo como un nodo de retransmisión. Esto significa que los robots intercambian toda la información del sistema como nodos de transporte, pero sólo procesan la información de la que son destinatarios.

6.2.7 Resultados experimentales.

Los resultados experimentales se muestran en esta sección. Para demostrar la escalabilidad del sistema propuesto, se propuso la realización de tareas con dos, tres y cuatro robots. Para incluir más robots, fue necesario añadir el reconocimiento de tramas de comunicación y un segmento de control para la nueva proyección. A nivel computacional, esto no incrementó significativamente el tiempo de procesamiento ya que se realizaron controladores dinámicos y cinemáticos en cada robot. Al final se incluyó un resultado simulado adicional para validar la escalabilidad del sistema de control. El esquema de control presentado en la Figura 6.25 permite cumplir los objetivos de forma y posición para un sistema de múltiples robots heterogéneos. Para la validación del controlador propuesto, se utilizaron robots de diferentes configuraciones

dinámicas para experimentar la escalabilidad del control cooperativo. En la Figura 6.30 se muestran los robots utilizados para los experimentos realizados. Se puede observar que el conjunto de robots era heterogéneo, con dos robots holonómicos omnidireccionales de diferentes características: El tipo A era un robot omnidireccional, mientras que el robot tipo B era muy similar, pero contenía componentes de mayor rendimiento que lo hacían capaz de soportar pesos pesados; el robot tipo C era un robot normal de uniciclo, con dos ruedas traseras de tracción, y finalmente el robot tipo D tenía una tracción a las cuatro ruedas.

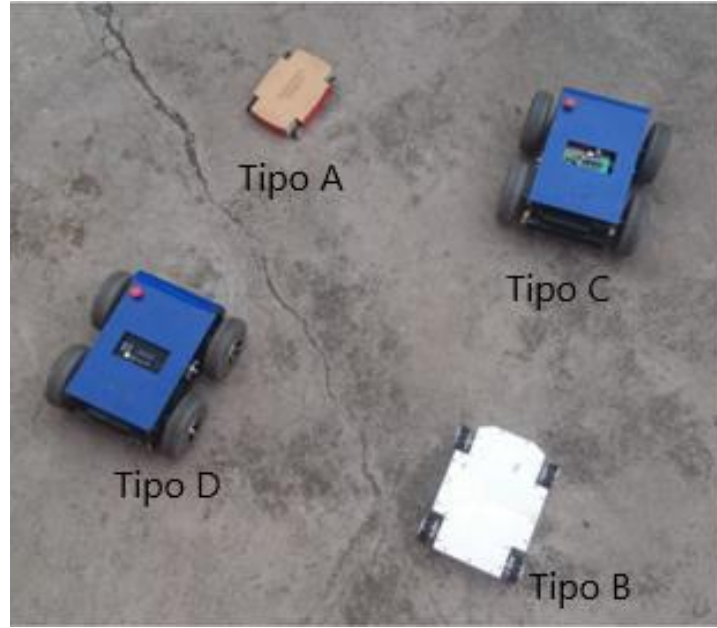


Figura 6.30. Todos los robots usados para las pruebas experimentales.

A. Primer experimento.

El primer experimento consistió en ejecutar una formación entre un robot uniciclo (robot tipo C) y un robot omnidireccional (robot tipo A). La trayectoria deseada $\rho(s)$ se describe con $x_{do} = \frac{4}{5} \cos\left(\frac{27}{100}t\right) + \frac{13}{100}t - 2$ [m], $y_{do} = \frac{3}{10}t - 4$ [m], durante 40 [seg] con $v_d = 0.2$ [$\frac{m}{s}$]; teniendo en cuenta que la trayectoria deseada puede ser el resultado de una planificación de trayectoria fuera de línea o planificada en línea. Los parámetros de formación definidos para el experimento son: $d_o = 1$ [m] y $\theta_o = \frac{\pi}{2} + \tan^{-1}\left(\frac{y_{do}}{x_{do}}\right)$ [rad]. La Figura 6.31 ilustra el camino ejecutado por los robots, donde el objetivo primario (x_o, y_o) permanece en el camino deseado. Por otro lado, el segundo objetivo durante el experimento cumple con la forma definida, es decir, la orientación y la distancia.

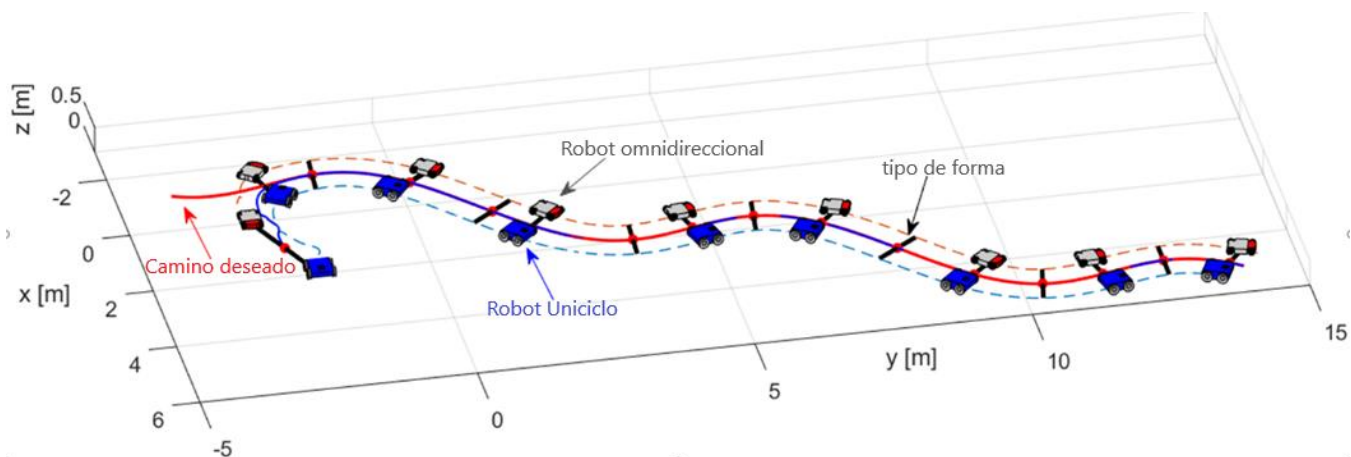


Figura 6.31 Movimientos ejecutados por los robots en 1er experimento

La Figura 6.32 presenta los errores de posición deseados $\tilde{\eta}(k)$ del objeto. Se observa que, al principio del experimento, hubo un error importante causado por las posiciones iniciales de cada robot. Una vez que se encontró el punto más cercano al camino, las plataformas móviles se posicionaron de tal manera que la posición del objetivo principal estaba en el camino deseado.

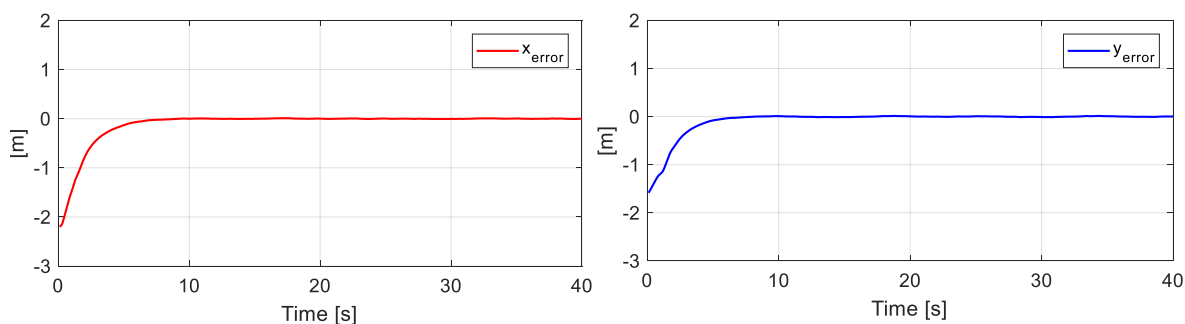


Figura 6.32 Errores del objetivo principal

La Figura 6.33 muestra el error del objetivo secundario, es decir, el error de distancia y el error de ángulo de formación entre ambos robots.

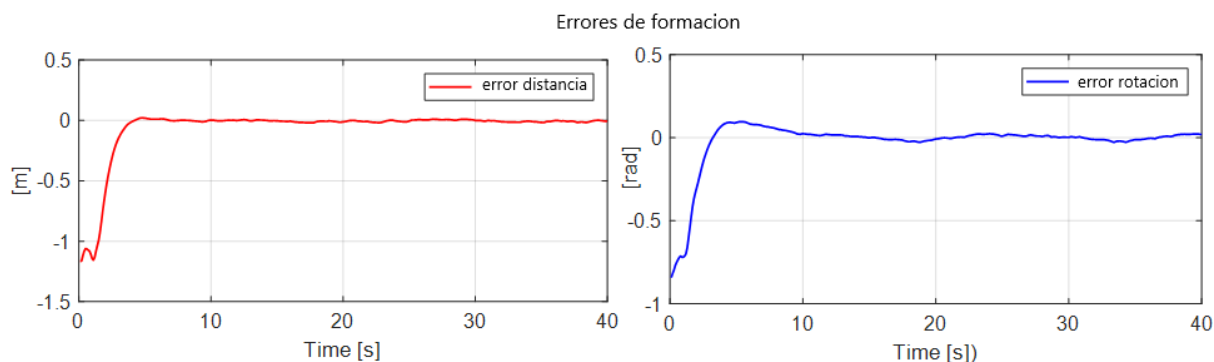


Figura 6.33 Errores del objetivo secundario

Para el experimento planeado, los controles de formación generaron referencias de velocidades $[\dot{h}_1 \ \dot{h}_2]$, que, por medio del control cinemático, cada robot llevó a seguir una trayectoria para cumplir con la formación esperada. Por lo tanto, en la Figura 6.34, se presentan los errores de posición de cada robot durante el experimento.

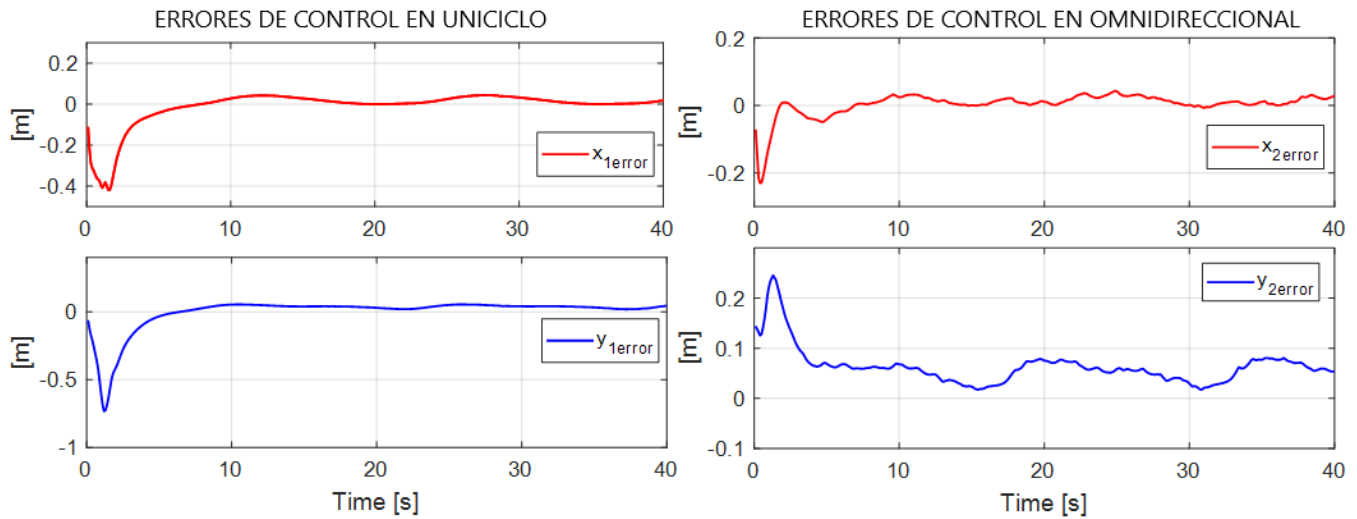


Figura 6.34 Errores de posición de los robots de unicycle y omnidireccionales durante el experimento.

B. Segundo Experimento.

El segundo experimento propone incorporar cuatro robots para la formación, dos plataformas móviles de tipo unicycle (robot tipo C y D) y dos plataformas móviles de tipo omnidireccional (robot tipo A y B). En contraste con el experimento anterior, la formación deseada para los cuatro robots varió los parámetros de formación durante el experimento, el cambio de forma consistió en variar el ángulo y las distancias virtuales definidas. La Figura 6.35 indica los parámetros de formación para el experimento de control cooperativo propuesto, el número de parámetros depende del tipo de forma y del número de robots incorporados. La Tabla 6.2 indica los parámetros deseados para este experimento.

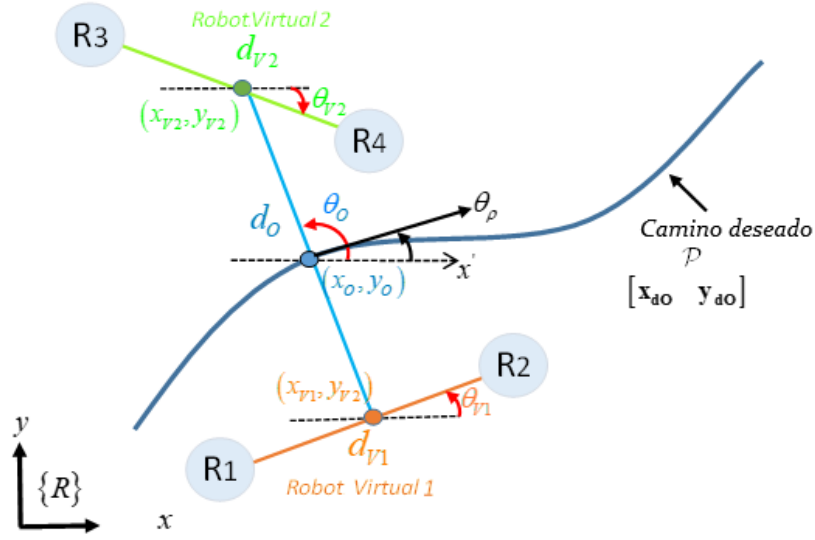


Figura 6.35 Esquema de los parámetros de formación para el segundo experimento.

Tabla 6.2 Parámetros de formación para el experimento

Parámetros de Formación	$0 > t < 30$ [s]	$30 \geq t \leq 50$ [s]
v_d	0.15 [m/s]	
x_{do}	$\frac{3}{10}t + 2$ [m]	
y_{do}	$\frac{2}{5}\sin\left(\frac{1}{5}t\right) + \frac{1}{40}t - 4$ [m]	
θ_ρ	$\tan^{-1}\left(\frac{\dot{y}_{do}}{\dot{x}_{do}}\right)$ [rad]	
d_o	1 [m]	2.5 [m]
θ_o	$\theta_\rho + \frac{\pi}{2}$ [rad]	$\tan^{-1}\left(\frac{\dot{y}_{do}}{\dot{x}_{do}}\right)$ [rad]
d_{v1}	1 [m]	2.5 [m]
θ_{v1}	$\tan^{-1}\left(\frac{\dot{y}_{do}}{\dot{x}_{do}}\right)$ [rad]	$\frac{\pi}{6}$ [rad]
d_{v2}	1 [m]	2 [m]
θ_{v2}	$\tan^{-1}\left(\frac{\dot{y}_{do}}{\dot{x}_{do}}\right)$ [rad]	$\frac{\pi}{6}$ [rad]

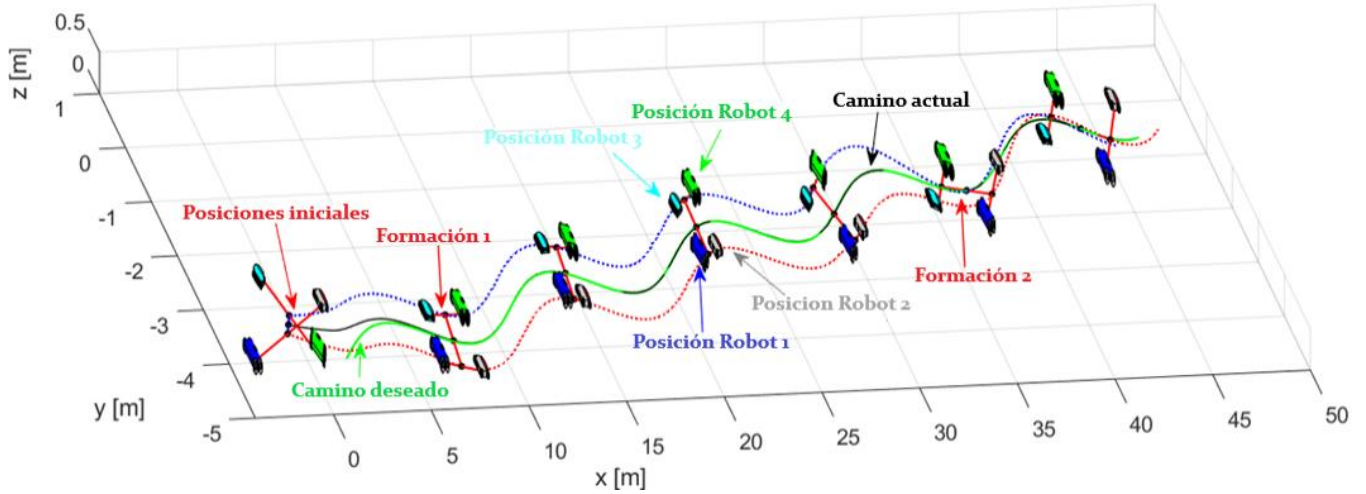


Figura 6.36 Movimiento de robots al completar la tarea.

A través del cambio de forma, se verificó que, durante la ejecución de la tarea, los robots podían modificar de forma autónoma las posiciones para mover el objeto de un lugar a otro siguiendo un camino definido. La Figura 6.36 muestra el movimiento estroboscópico de la tarea ejecutada, mientras que en la Figura 6.37, se aprecia cómo la forma del conjunto robótico cambia gradualmente.

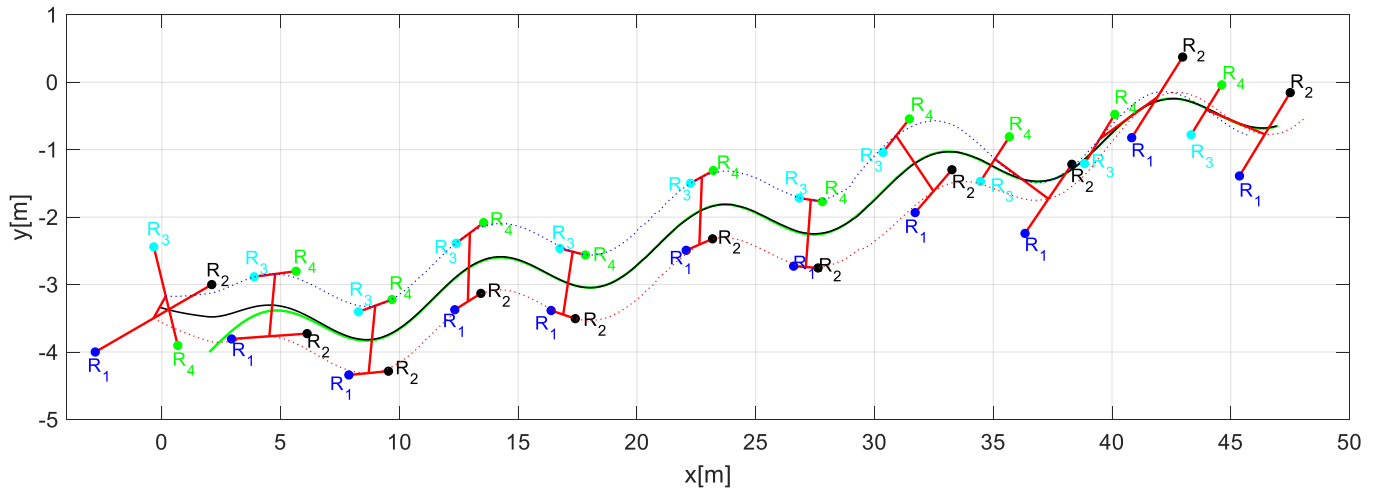


Figura 6.37 Movimiento estroboscópico de los robots en el segundo experimento

En la Figura 6.38 se muestran los errores de la formación (objetivo principal - objetivos secundarios), es decir, los errores del punto principal $[\tilde{x}_o, \tilde{y}_o, \tilde{d}_o, \tilde{\theta}_o]$. De la misma manera, los errores de los parámetros virtuales se denotan como $[\tilde{x}_{v1}, \tilde{y}_{v1}, \tilde{d}_{v1}, \tilde{\theta}_{v1}, \tilde{x}_{v2}, \tilde{y}_{v2}, \tilde{d}_{v2}, \tilde{\theta}_{v2}]$.

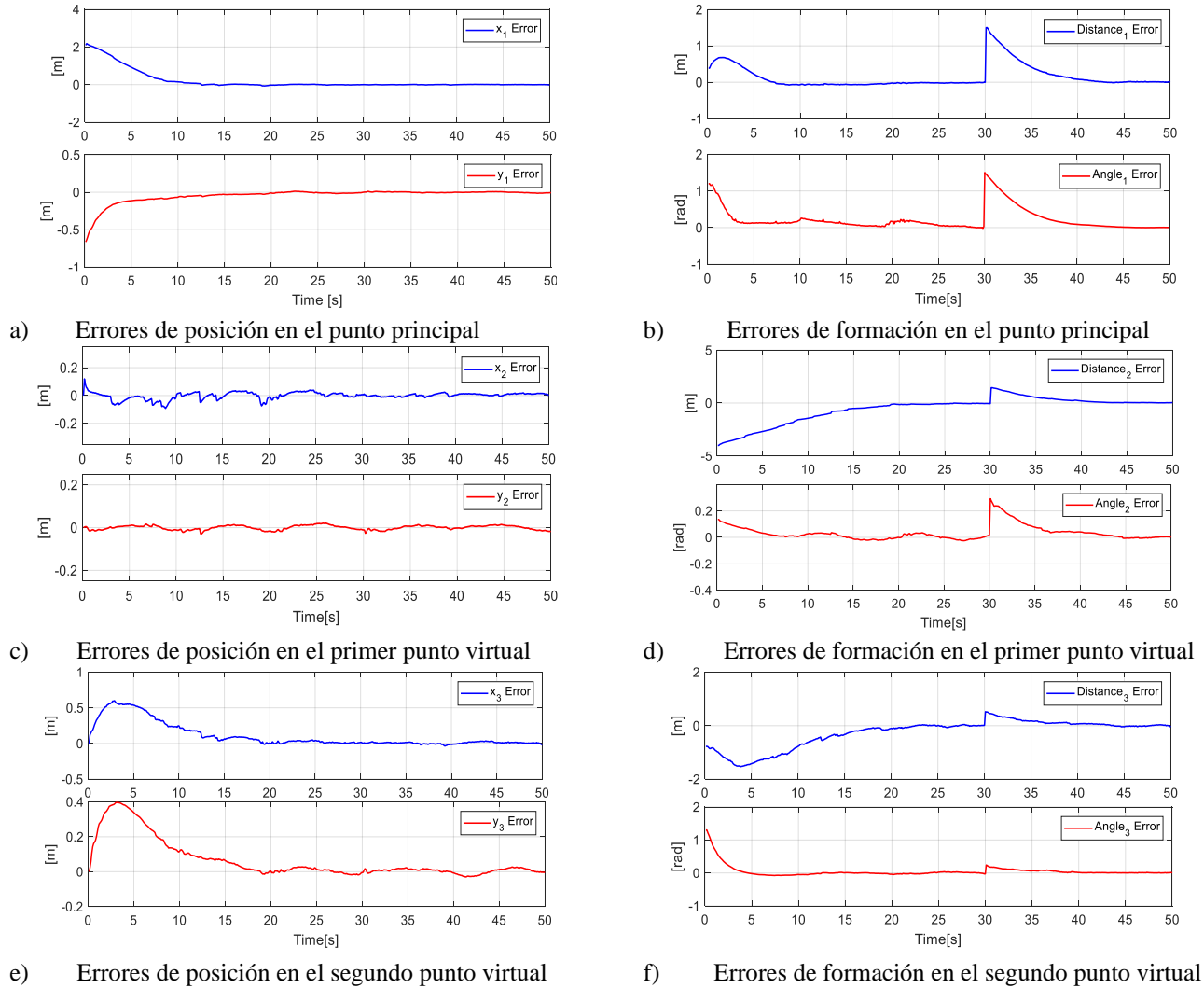
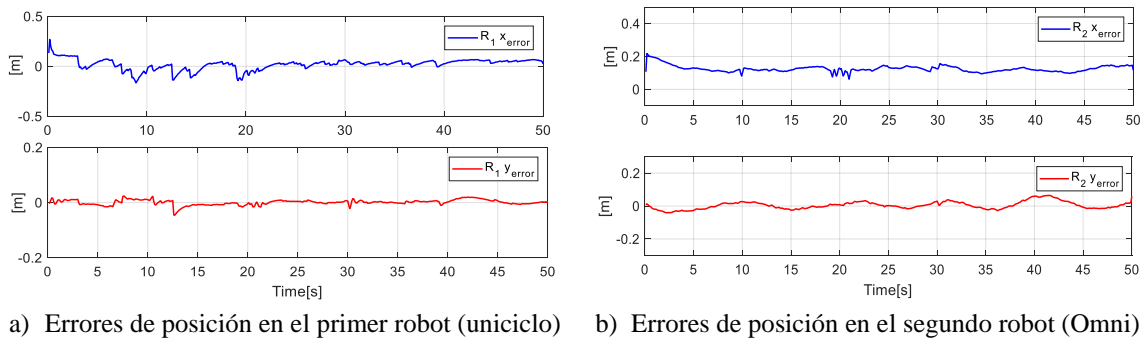
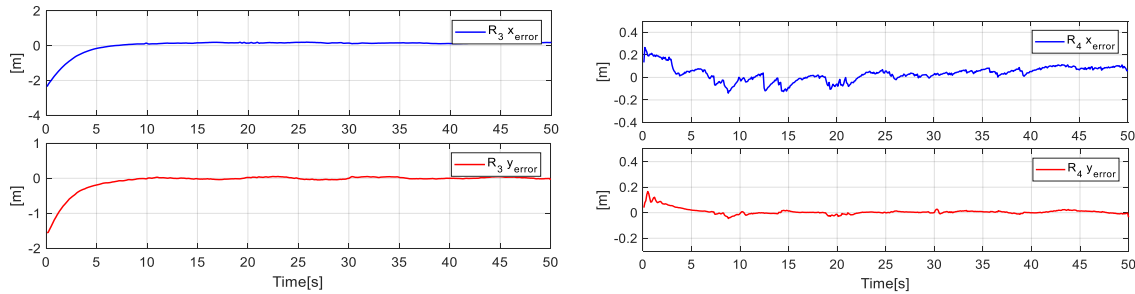


Figura 6.38 Errores de objetivo principal y errores de objetivo secundario en el punto primario y puntos virtuales de la formación.

Por último, la Figura 6.39 indica los errores de posición de cada robot durante la experimentación, donde se muestra que el error se aproxima a cero de cada robot en la experimentación ejecutada.





c) Errores de posición en el tercer robot (Omni) d) Errores de posición en el cuarto robot (Uniciclo)

Figura 6.39 Los errores de posición de cada uno de los cuatro robots para la forma deseada.

Con el fin de verificar el alcance y escalabilidad de los algoritmos de colaboración se hicieron tres experimentos más que no están consignados en este trabajo, además de simulaciones con hasta 8 y 9 (dadas las limitaciones del hardware), no han sido considerados por la extensión que tiene; sin embargo, los resultados arrojan las mismas características que en estos dos experimentos se pueden observar, resultados que demuestran el correcto funcionamiento del controlador propuesto a través de diferentes experimentos reales en los que se utilizan hasta cuatro mecanismos diferentes.

Todo lo descrito puede aplicarse en entornos de construcción civil, para el transporte y la colocación de materiales en lugares estratégicos; en la gestión de almacenes para el transporte de objetos pesados de diversas características y dimensiones, y finalmente también en aplicaciones militares para la colocación de objetivos disuasorios en campos de entrenamiento.

Finalmente, aunque no se aborde todas las aplicaciones publicadas como productos de esta investigación, se aborda con el último desarrollo, el final del objetivo específico 1, que considera una parte importantísima del trabajo de robots autónomos: la autonomía energética, así como su gestión; y las formas de ahorrar las fuentes poco disponibles de energía que consume un robot móvil.

6.3 TERCERA APLICACIÓN: Desarrollo de la aplicación de ahorro de energía para tareas de manipuladores móviles.

Esta aplicación presenta el control cinemático y dinámico de un sistema robótico manipulador móvil basado en métodos numéricos. Además, presenta el análisis de curvatura de un camino no parametrizado en el tiempo, para la optimización del consumo de energía. La optimización de la energía considera dos aspectos: la velocidad de ejecución en las curvas y la cantidad de movimientos generados por el sistema robótico. Cuando se produce una curva en la trayectoria predefinida, la velocidad de ejecución se analiza en todo el sistema en un método unificado para evitar que los efectos del deslizamiento afecten al manipulador móvil, mientras que el número

de movimientos está limitado por la redundancia que presenta el sistema robótico para optimizar el consumo de energía. Los resultados experimentales se muestran para validar la construcción mecánica y electrónica del sistema, los controladores propuestos y el ahorro de consumo de energía.

En esta aplicación, se analizan características como la redundancia y la velocidad de movimiento del sistema de manipulador robótico móvil a través de métodos numéricos para plantear optimizadores de energía, una característica muy importante y tratada en algunos trabajos. Con el fin de alcanzar los objetivos de ahorro de energía y evitar el consumo innecesario de ella, se desarrolla el análisis del radio de curvatura de una carretera no parametrizada en el tiempo, considerando el cálculo de las velocidades adecuadas para determinados períodos de tiempo. Asimismo, se tiene en cuenta y se analiza la consideración de reducir el movimiento de la plataforma móvil en comparación con el movimiento del brazo robótico. Se desarrolla un modelo dinámico para el manipulador móvil que, a diferencia de trabajos anteriores, acepta entradas de velocidad como es habitual en los robots comerciales, donde se comprueba la estabilidad tanto de la propuesta cinemática como del controlador dinámico a través de las propiedades del álgebra lineal. Para validar los algoritmos de control propuestos y la optimización del costo de la energía propuesta, se incluyen y discuten los resultados experimentales. Además, se desarrolla la construcción mecánica y electrónica de un brazo de 6 grados montado en una de las plataformas antes descritas y diseñadas, la plataforma móvil tipo uniclo, que tiene una computadora incorporada para la ejecución del controlador cinemático y dinámico, junto con la propuesta de ahorro de energía.

6.3.1 Propuesta de ahorro de energía.

El consumo de energía del robot rara vez se tiene en cuenta al alcanzar los puntos deseados de una trayectoria. La manipulación de la velocidad de desplazamiento de un robot permite un uso más eficaz de la energía, asimismo, en los sistemas robóticos compuestos por varios robots (manipulador móvil sobre ruedas, manipulador aéreo, robot hexápodo, robots humanoides, y así sucesivamente) también es posible optimizar el consumo de energía a través de esquemas de control que permiten alcanzar el objetivo deseado con un menor número de movimientos, logrando esta optimización a través de restricciones en el control de la redundancia del sistema.

La velocidad de desplazamiento en la ejecución del seguimiento de la trayectoria está directamente relacionada con el consumo de energía. Analizando el problema como una ejecución con velocidad constante en un camino que incluye curvas, la pérdida de energía podría producirse principalmente por efectos de deslizamiento y corrección de las fuerzas centrífugas y centrípetas resultantes, de acuerdo con la configuración del robot que está ejecutando una

tarea específica. Además, cabe señalar que el consumo de energía de todo el sistema equivale a la suma de las potencias consumidas en cada uno de los actuadores del robot.

Así, en el caso de un robot manipulador compuesto por una plataforma móvil con j_p actuadores (P_{jp}) y un brazo robótico j_a con actuadores (P_{ja}), la potencia total del sistema robótico se define como $P_T = P_{1p} + P_{2p} + \dots + P_{mp} + P_{1a} + P_{2a} + \dots + P_{na}$, la misma que equivale al consumo de todo el sistema en un período de tiempo definido. Como se ha descrito anteriormente, el gasto de energía puede determinarse por: $E_T = \int_{T_0}^{T_F} P_T(t)dt$ donde T_0 y T_F son la hora de inicio y fin de la ejecución de la tarea del sistema robótico.

Observación 1: Según la configuración del brazo robótico, la potencia de cada uno de los actuadores dependerá de la masa y el tamaño del eslabón o eslabones a desplazar; mientras que la potencia consumida por los actuadores de la plataforma móvil es igual entre ambos actuadores, dado que la potencia necesaria para desplazar una carga se distribuye entre cada uno de los actuadores.

Observación 2: La potencia necesaria para desplazar todo el sistema robótico requiere un mayor consumo en los motores de la plataforma móvil en comparación con los actuadores del brazo.

En el dimensionamiento del consumo de energía se estima que el costo del consumo de cada uno de los actuadores de la plataforma móvil es hasta cuatro veces mayor que el consumo de las dos primeras articulaciones del brazo robótico (considerando que los dos primeros motores del brazo robótico son los más potentes, en comparación con el resto de los actuadores del manipulador). Por lo tanto, el ahorro de energía de un manipulador móvil al ejecutar una tarea específica es directamente proporcional al número de movimientos realizados por la plataforma y el brazo simultáneamente.

Basándose en lo anterior, este trabajo propone un esquema de control que optimiza el consumo de energía de un robot manipulador móvil al ejecutar una tarea específica. Como criterio de optimización energética se considera: i) la manipulación de la velocidad deseada para ejecutar la tarea; y ii) la reducción del número de movimientos del manipulador móvil, dando mayor importancia al movimiento del brazo con respecto al de la plataforma móvil.

6.3.2 Modelo del manipulador móvil.

La configuración del manipulador móvil está definida por un vector q de n_a coordenadas independientes llamadas coordenadas generalizadas del manipulador móvil con $q = [q_1 \ q_2 \ \dots \ q_n]^T = [q_p^T \ q_a^T]^T$, donde q_a representa las coordenadas generalizadas del brazo y q_p las coordenadas generalizadas de la plataforma móvil. Se observa que $n = n_a + n_p$, donde n_a y n_p son las dimensiones de los espacios generalizados asociados al brazo robótico y a la plataforma móvil respectivamente.

La configuración q es un elemento del espacio de configuración del manipulador móvil; denotado por \mathcal{N} . La ubicación del efector final del manipulador móvil viene dada por el vector m -dimensional $h = [h_1 \ h_2 \ \dots \ h_m]^T$, donde $h(t)$ define la posición y la orientación del efector final del manipulador móvil en \mathcal{R} . Sus coordenadas m son las coordenadas operacionales del manipulador móvil. El conjunto de todas las posiciones constituye el espacio operativo del manipulador móvil, denotado por \mathcal{M} .

a. Modelo cinemático.

El modelo cinemático de un manipulador móvil da la ubicación del efector final h en función de la configuración del brazo robótico y la ubicación de la plataforma (o sus coordenadas operacionales en función de las coordenadas generalizadas del brazo robótico y las coordenadas operacionales de la plataforma móvil).

$$f: \mathcal{N}_a \times \mathcal{M}_p \rightarrow \mathcal{M}$$

$$(q_a, q_p) \mapsto h = f(q_a, q_p)$$

donde, \mathcal{N}_a es el espacio de configuración del brazo robótico, \mathcal{M}_p es el espacio operacional de la plataforma.

El modelo cinemático instantáneo de un manipulador móvil da la derivación de su ubicación del efector final en función de las derivaciones tanto de la configuración del brazo robótico como de la ubicación de la plataforma móvil. Ahora, después de analizar las afirmaciones anteriores, podemos representar la velocidad del efector final como sigue:

$$\dot{h}(t) = J(q)v(t) \tag{6.60}$$

donde, $J(q)$ es la matriz jacobiana que define un mapeo lineal entre el vector de las velocidades del manipulador móvil $v(t)$ y el vector de la velocidad del efector final $\dot{h}(t)$. La matriz jacobiana es, en general, una función de la configuración $q(t)$. Aquellas configuraciones en las que $J(q)$ es deficiente en rango se denominan configuraciones cinemáticas singulares. Encontrar las singularidades del manipulador es de gran interés debido a las siguientes razones principales: a) Las singularidades representan configuraciones en las que la movilidad de la estructura es reducida (es decir, no es posible imponer un movimiento arbitrario al efector final); b) En la vecindad de una singularidad, pequeñas velocidades en el espacio operativo pueden causar grandes velocidades en el espacio $q(t)$.

Observación 3: En general, la dimensión del espacio operativo m es menor que el grado de movilidad δ_n del manipulador móvil. Por lo tanto, hay un problema de redundancia que debe ser resuelto en el control de las tareas del robot manipulador móvil.

b. Modelo dinámico.

El modelo matemático que representa la dinámica de un manipulador móvil puede obtenerse a partir de las ecuaciones dinámicas de Lagrange, que se basan en la diferencia entre la energía cinética y la energía potencial de cada una de las articulaciones del robot (balance de energía). La mayoría de los robots disponibles en el mercado tienen controladores PID de bajo nivel para seguir las entradas de velocidad de referencia, lo que no permite controlar los motores directamente. Por lo tanto, resulta útil expresar más adecuadamente el modelo dinámico del manipulador móvil considerando las velocidades de referencia rotacional y longitudinal como las señales de entrada. Para ello, se incluyen en el modelo los controladores de velocidad.

$$M(q)\dot{v} + C(q, v)v + g(q) + d = v_{ref} \quad (6.61)$$

donde

$$M(q) = H^{-1}(\tilde{M} + D),$$

$$C(q, v) = H^{-1}(\tilde{C} + P),$$

$$g(q) = H^{-1}\tilde{g}(q),$$

$$d = H^{-1}\tilde{d}$$

$$v_{ref} = [u_{ref} \quad \omega_{ref} \quad \dot{\theta}_{1ref} \quad \dot{\theta}_{2ref} \quad \dots \quad \dot{\theta}_{na_{ref}}]^T.$$

Así, $M(q) \in \mathbb{R}^{\delta_n \times \delta_n}$ es una matriz definida positiva que representa la inercia del sistema, $C(q, v)v \in \mathbb{R}^{\delta_n}$ representa las fuerzas centrípeta y de Coriolis, $g(q) \in \mathbb{R}^{\delta_n}$ es el vector gravitatorio, d denota perturbaciones desconocidas delimitadas, incluida la dinámica no modelada y $v_{ref} \in \mathbb{R}^{\delta_n}$ es el vector de las señales de control de la velocidad, $H \in \mathbb{R}^{\delta_n \times \delta_n}$, $D \in \mathbb{R}^{\delta_n \times \delta_n}$ y $P \in \mathbb{R}^{\delta_n \times \delta_n}$ son matrices diagonales positivas definidas y constantes que contienen los parámetros físicos del manipulador móvil, los motores y los controladores de velocidad tanto de la plataforma móvil como del manipulador.

Es importante notar que el modelo dinámico del manipulador móvil puede ser representado por:

$$M(q)\dot{v} + C(q, v)v + g(q) + d = \Phi(q, v, \sigma)\chi$$

donde, $\Phi(q, v, \sigma) \in \mathbb{R}^{\delta_n \times l}$ y $\chi = [\chi_1 \ \chi_2 \ \cdots \ \chi_l]^T$ es el vector de parámetros desconocidos del manipulador móvil (es decir, masa del robot móvil, masa del brazo robótico, parámetros físicos del manipulador móvil, motores, velocidad, etc.).

El modelo matemático completo del robot manipulador móvil está representado por: (6.60) el modelo cinemático instantáneo y (6.61) el modelo dinámico, tomando las velocidades de referencia del sistema como señales de entrada.

6.3.3 Diseño del controlador.

Los criterios de diseño del control propuesto consideran la manipulación de la velocidad deseada para el seguimiento de caminos, así como el control de redundancia del sistema del manipulador móvil. La propuesta de control se aplica para seguir caminos no parametrizados en el tiempo, porque la velocidad deseada puede modificarse en función de la tarea a ejecutar por el sistema robótico. El esquema de control propuesto para resolver el problema del control del movimiento se muestra en la Figura 6.40, donde el diseño del controlador se basa en dos subsistemas en cascada:

i) Un controlador cinemático de norma mínima con saturación de comandos de velocidad, donde sus entradas son $h_d(t, s, h)$ y $v_{hd}(t, s, h)$, que describen la ubicación deseada y la velocidad del efector final del manipulador móvil. El error de control viene dado por el error de ubicación del efector final definido como $\tilde{h} = h_d - h$. Por lo tanto, el objetivo de control se expresa como:

$$\lim_{t \rightarrow \infty} \tilde{h}(t, s, h) = 0 \in \mathbb{R}^m$$

ii) Controlador de compensación dinámica, cuyo objetivo principal es compensar la dinámica del manipulador móvil, reduciendo así el error de seguimiento de la velocidad. Este controlador recibe como entradas las velocidades deseadas calculadas por el controlador cinemático, y genera referencias de velocidad v_{ref} para el robot manipulador móvil. El error de control de velocidad se define como $\tilde{v} = v_c - v$. Por lo tanto, el objetivo del control es asegurar que:

$$\lim_{t \rightarrow \infty} \tilde{v}(t) = 0 \in \mathbb{R}^{\delta_n}$$

Observación 4: Ambos controladores se basan en métodos numéricos.

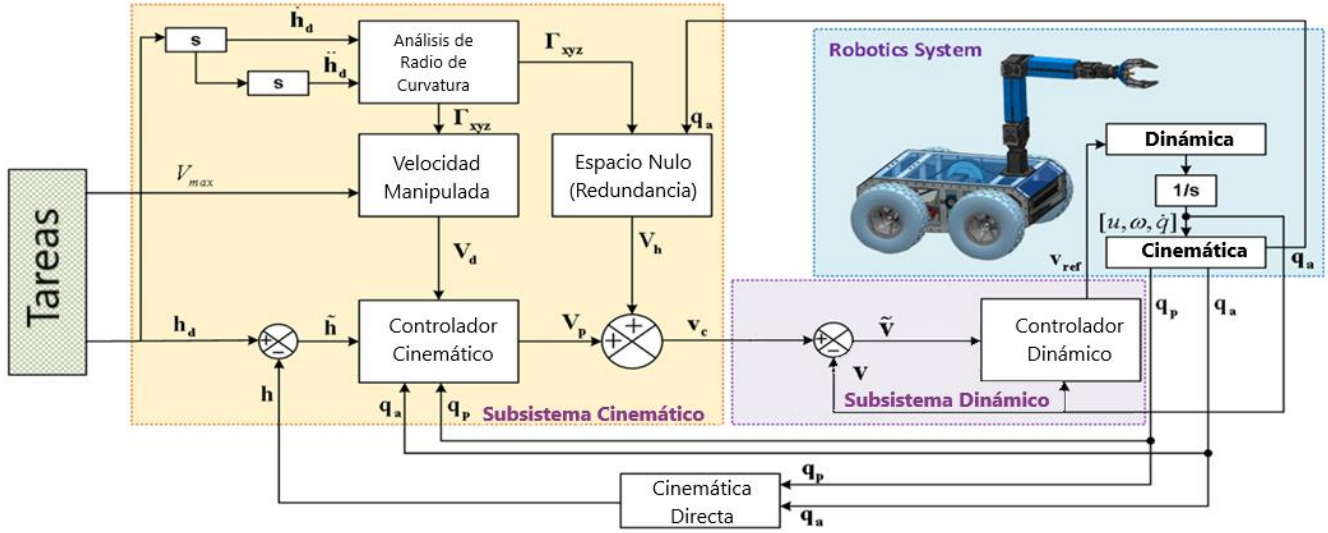


Figura 6.40. Diagrama de bloques de los problemas de control de movimiento de los manipuladores móviles.

a. Controlador cinemático.

El controlador cinemático evalúa los errores de control en cada período de muestreo, de manera que la velocidad real obtenida del manipulador móvil puede utilizarse como referencia para maniobrar todo el sistema, reduciendo así los errores. En esta sección se presenta el seguimiento del camino considerando la manipulación de una velocidad deseada con el fin de reducir el consumo de energía, donde la tarea a ejecutar estará en función del camino seguido y de la velocidad deseada.

A. Especificación del camino.

La Figura 6.41 muestra el problema de rastreo de trayectoria representado por $\mathcal{P}(s)$, donde $\mathcal{P}(s) = (x_p(s), y_p(s), z_p(s))$; \mathcal{P}_d representa el punto actual deseado del robot manipulador móvil que se considera el punto más cercano $\mathcal{P}(s)$ al sistema robótico, esto se define como $\mathcal{P}_d = (x_p(s_d), y_p(s_d), z_p(s_d))$, donde s_d es la abscisa curvilínea definida del punto \mathcal{P}_d ; la representación de los errores en dirección \mathcal{X} viene dada por $\tilde{x} = x_p(s_d) - x$; la representación de los errores en dirección \mathcal{Y} viene dada por $\tilde{y} = y_p(s_d) - y$; la representación de los errores en dirección \mathcal{Z} viene dada por $\tilde{z} = z_p(s_d) - z$.

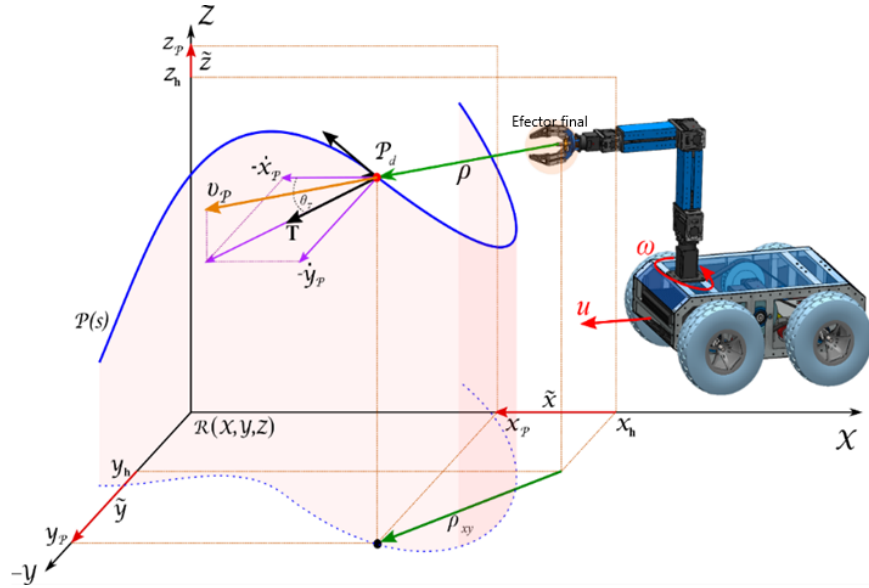


Figura 6.41 Seguimiento de caminos

Basándose en la representación gráfica, los errores de control $\rho(t)$ se calculan por la diferencia de posición entre el extremo operativo $h(x, y, z)$ y el punto deseado \mathcal{P}_d , donde la distancia entre la posición actual del robot $h(x, y, z)$ y el punto de referencia es cero $\tilde{\rho} = 0 - \rho = -\rho$; $\tilde{\psi}(t)$ es el error de orientación del efector final definido como $\tilde{\psi} = \theta_T - \psi$, donde θ_T es la orientación del vector unitario tangente a la trayectoria del punto \mathcal{P}_d con respecto al sistema de referencia $R(X, Y, Z)$.

Observación 4: La orientación del efector final depende directamente de la aplicación, por lo que no se estudia en este trabajo, porque el objetivo del algoritmo es que el efector alcance el punto de interés con un menor consumo de energía. Por ejemplo, la aplicación puede ser tomar una botella (orientación vertical), tirar de una hoja de papel (orientación horizontal), o cualquier otra orientación de la pinza. Esto no afecta realmente al algoritmo, ya que será el mismo para cualquier aplicación.

B. Definición de la velocidad deseada.

La consideración del ahorro de energía propone como primera instancia la reducción de la velocidad en las curvas al ejecutar el seguimiento de una carretera, por lo tanto, la velocidad deseada puede ser manipulada. Para el seguimiento de una ruta, la velocidad deseada puede depender de diferentes parámetros, por ejemplo, la tarea a ejecutar, los errores de control, la curvatura de la ruta, los criterios de optimización, entre otros.

$$v_d(t) = f(s, q, \Gamma)$$

Con el fin de optimizar el consumo de energía en la ejecución de una tarea deseada, el esquema de control propuesto en este documento considera que la velocidad de referencia en el camino a seguir es una función de su radio de curvatura.

$$v_d(t) = \frac{v_{max}}{1 + k \tanh(\Gamma(t))}$$

donde Γ representa el radio de curvatura y k es una constante adimensional. Al considerar el camino requerido p como un conjunto de puntos, el valor de la curvatura se define como:

$$\Gamma(k) = \frac{|\dot{p}(k) \times \ddot{p}(k)|}{|\dot{p}(k)|^3} \quad (6.62)$$

Observación 5: Los valores de la curvatura del radio en cada instante del tiempo de (6.62) sólo se pueden encontrar si se tiene la expresión analítica del camino. Esto limita en gran medida el uso de este tipo de consideración, ya que para las aplicaciones reales no siempre existe el camino a seguir en forma de ecuaciones matemáticas derivables.

Para resolver la limitación de no tener la expresión analítica, se propone utilizar los puntos $p(k + 1)$ siguientes y $p(k - 1)$ anteriores del período de muestreo, de esta manera $\dot{p}(k)$, se define de la siguiente manera:

$$\dot{p}(k) = \frac{p(k - 1) - p(k + 1)}{2T_s}$$

y el valor $\ddot{p}(k)$ se calcula por:

$$\ddot{p}(k) = \frac{p(k + 1) - 2p(k) + p(k - 1)}{T_s^2}$$

El análisis del radio de curvatura puede desarrollarse por separado, ya que la plataforma móvil se mueve en el plano x, y , mientras que el brazo en el espacio x, y, z .

C. Diseño de la ley de control.

El diseño de los controladores cinemáticos y dinámicos propuestos en este trabajo se basan en herramientas de métodos numéricos. Concretamente, para la solución de sistemas de ecuaciones, estos sistemas pueden ser representados en forma matricial, para lo cual se aplican teoremas y axiomas del álgebra lineal:

Considerando la ecuación diferencial de primer orden:

$$\dot{h}(t) = f(h, v, \Gamma) \text{ con } h(0) = h_0$$

donde h representa la salida del sistema a ser controlador; \dot{h} es la primera derivada con respecto al tiempo; v es la acción de control; Γ representa diferentes criterios de la tarea a ejecutar. Los valores de $h(t)$ en el tiempo discreto $t = kT_0$ se denominan $h(k)$, donde T_0 representa el tiempo de muestreo y $k \in \{1,2,3,4,5 \dots\}$. Además, el uso de métodos numéricos para el cálculo de la evolución del sistema se basa principalmente en la posibilidad de aproximar el sistema de estados en el instante de tiempo k , si se conoce el estado y la acción de control en el instante de tiempo $k - 1$, esta aproximación se denomina método de Euler.

$$\frac{h(k) - h(k - 1)}{T_0} = f(h, v, \Gamma) \quad (6.63)$$

El diseño del controlador cinemático se basa en el modelo cinemático del robot manipulador móvil. Para diseñar el controlador cinemático, el modelo de robot (6.60) puede ser aproximado como (6.63).

$$\frac{h(k) - h(k - 1)}{T_0} = J(q(k))v(k) \quad (6.64)$$

Cabe señalar que el rastreo de la trayectoria consiste en mantener el extremo operativo del manipulador móvil dentro de una ruta predefinida sin parametrización del tiempo. De esta manera, el objetivo del control es posicionar el punto de interés en el punto más cercano de la ruta p a una velocidad deseada v_d . Para lograr este objetivo, se considera la siguiente expresión:

$$\frac{h(k) - h(k - 1)}{T_0} = v_d(k) + W \left(\frac{h_d(k - 1) - h(k - 1)}{T_0} \right) \quad (6.65)$$

donde, h_d es el camino deseado, $W(\tilde{h}(k - 1))$ es una matriz diagonal que controla los pesos de los errores, definidos como:

$$W(\tilde{h}_m(k - 1)) = \frac{w_m}{1 + |\tilde{h}_m(k - 1)|}$$

donde m representa las coordenadas operacionales del robot manipulador móvil.

Ahora, para generar las ecuaciones del sistema considere (6.64) y (6.65), y el sistema puede ser reescrito como $Au = b$

$$\underbrace{J(q(k))}_A \underbrace{v(k)}_u = \underbrace{v_d(k) + \frac{W(h_d(k - 1) - h(k - 1))}{T_0}}_b \quad (6.66)$$

Observación 6: Considerando que la configuración del sistema robótico es redundante (6.66), la matriz jacobiana $J \in R^{m \times n}$ tiene más incógnitas que ecuaciones ($m < n$), con un rango $r = n$ para cada $b \in R^m$, entonces (6.66) representa un sistema lineal subdeterminado con solución general.

$$v = v_p + v_h \quad (6.67)$$

donde v_p es una solución particular y v_h es una solución de sistema homogéneo $Jv_h = 0$. Este resultado es de fundamental importancia para la resolución de sistemas redundantes.

Un método de solución viable es formular el problema como un problema de optimización lineal restringida.

$$\frac{1}{2} \|v\|_2^2 = \min.$$

Entonces da la solución particular:

$$v_p = J^T(JJ^T)^{-1}b \quad (6.68)$$

Por otro lado, el espacio nulo de J en R^n es el conjunto de velocidades conjuntas y velocidades de la plataforma móvil que no producen ninguna velocidad de efecto final del manipulador móvil.

En ese caso, es necesario considerar un nuevo costo funcional en la forma:

$$\frac{1}{2} \|v - v_0\|_2^2 = \min.$$

Entonces se obtiene la solución homogénea

$$v_h = (I_n - J^T(JJ^T)^{-1}J)v_0 \quad (6.70)$$

Por lo tanto, este trabajo propone la inclusión de dos tipos de soluciones, la solución particular para optimizar las velocidades de rastreo del camino de todo el sistema robótico y la solución homogénea para reducir la cantidad de movimientos de la plataforma móvil con respecto al brazo robótico.

De esta manera, al insertar la expresión (6.68) y (6.70) en (6.67) se obtiene la solución en lugar de representar la ley de control propuesta.

$$v_c = \underbrace{J^T(JJ^T)^{-1}b}_{v_p} + \underbrace{(I_n - J^T(JJ^T)^{-1}J)v_0}_{v_h} \quad (6.71)$$

donde el primer término de la izquierda es la solución particular (v_p), y el segundo término (v_h) de esta ecuación pertenece al espacio nulo J .

El segundo término en (6.71) representa la proyección en el espacio nulo del manipulador móvil, donde v_0 es un vector arbitrario que contiene las velocidades asociadas al manipulador móvil. Por lo tanto, cualquier valor dado a v_0 afectará sólo a la estructura interna del manipulador, pero no al control final del efector en absoluto, como se muestra en la Figura 6.42. Utilizando este término se pueden lograr diferentes objetivos de control, por ejemplo, máxima manipulabilidad, ahorro de energía, evitación de obstáculos, distancia de los límites de las juntas mecánicas, etc. En este trabajo se considera como objetivo secundario que el sistema de manipulador robótico móvil debe ejecutar la tarea con una configuración de máxima manipulabilidad y al mismo tiempo con un criterio de optimización de la energía.

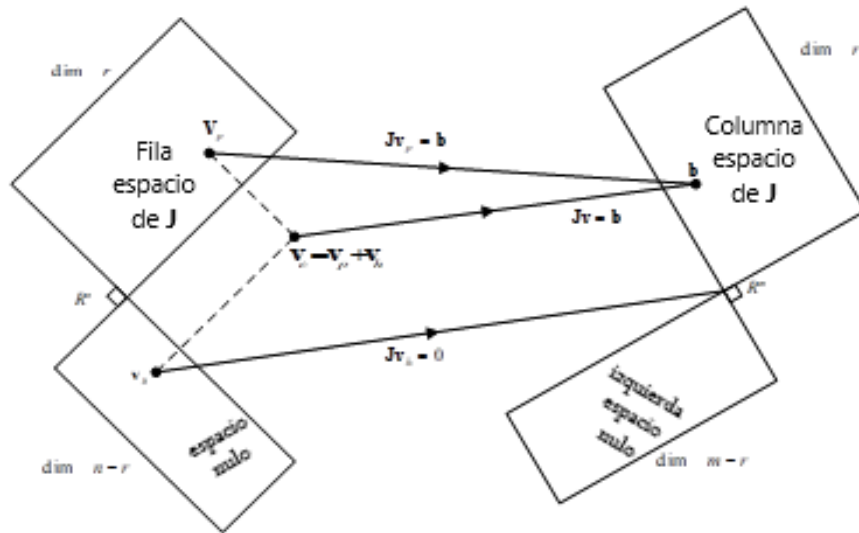


Figura 6.42. Proyecciones ortogonales de operación

D. Objetivos secundarios

Podemos observar que uno de los principales requisitos para una ejecución precisa de la tarea por parte del robot es una buena manipulabilidad, definida como la configuración del robot que maximiza su capacidad para manipular un objeto objetivo. Por lo tanto, uno de los objetivos secundarios del control es mantener la máxima manipulabilidad del manipulador móvil durante la ejecución de la tarea. La manipulabilidad es un concepto introducido por Yoshikawa (1985) para medir la capacidad de un manipulador fijo para moverse en ciertas direcciones. Bayle y Fourquet (2001) presentan un análisis similar para la manipulabilidad de los manipuladores móviles y amplían el concepto de elipsoide de manipulabilidad como el conjunto de todas las velocidades del efector final alcanzables por las velocidades de los robots v satisfaciendo $\|v\| \leq 1$ en el espacio euclidiano. Se puede obtener una medida representativa global de la capacidad de manipulación considerando el volumen de este elipsoide que es proporcional a la cantidad w , llamada medida de manipulabilidad,

$$w = \sqrt{\det(J(q)J^T(q))} \quad (6.72)$$

En este trabajo se considera la máxima manipulabilidad del sistema robótico para evitar configuraciones no deseadas (por ejemplo, $J^\# \rightarrow \infty$), además, la redundancia del sistema se utiliza para fines de optimización de la energía. Para cumplir estos objetivos secundarios, se considera que el vector arbitrario v_0 es una función de la manipulabilidad del robot, la velocidad deseada y el radio y dirección de curvatura de la carretera.

$$v_0 = g(w, v_{max}, \Gamma, \text{sign}(\Gamma))$$

Por lo tanto, v_0 debe permitir que el brazo robótico realice el mayor número de movimientos en relación con la plataforma móvil, sin que el brazo robótico tenga nunca una configuración única, es decir, cuando la carretera no tiene curvas, el sistema robótico se posiciona en la configuración de máxima manipulabilidad, mientras que cuando hay curvas, el brazo tiene mayor libertad de movimiento para mantener la trayectoria deseada. Por lo tanto, el vector arbitrario se define como:

$$v_0 = [u_0 \quad \omega_0 \quad q_{1_0} \quad q_{2_0} \quad \cdots \quad q_{na_0}]^T \quad (6.73)$$

Las velocidades que afectan a la plataforma móvil afectan a la posición del extremo operativo del sistema robótico en el plano x,y del sistema de referencia $R(X, Y, Z)$, por lo tanto,

$$u_0 = f \left(v_{max}, \Gamma_{xy} \frac{u_{max}}{1 + k_u \tanh(l_u \Gamma_{xy}(k))} \right)$$

$$\omega_0 = f(\Gamma_{xy}, \zeta_{xy}) = \zeta_{xy}(k) k_\omega \tanh(l_\omega \Gamma_{xy}(k))$$

donde $\Gamma_{xy}(k)$ es el producto cruz resultante de la velocidad y la aceleración en las curvaturas proyectadas en los planos x,y (se anulan tanto las velocidades como las aceleraciones de la tercera componente) y se define como

$$\Gamma_{xy}(k) = \frac{|\dot{r}(k) \times \ddot{r}(k)|}{|\dot{r}(k)|^3}; \text{ con } r(k) = [h_x(k) \quad h_y(k) \quad 0]^T \quad (6.74)$$

$\zeta_{xy} = \text{sign}(\dot{r}(k) \times \ddot{r}(k))$ es la dirección de la curvatura en el plano; x,y y $k_u, l_u, k_\omega, l_\omega$ son constantes de ganancia que ponderan la incidencia de la curvatura del camino en función del movimiento de la plataforma robótica.

A diferencia, los componentes $q_{1_0} \cdots q_{na_0}$ consideran la configuración interna del brazo robótico y el desplazamiento del extremo operativo en el espacio x,y,z del sistema de referencia $R(X, Y, Z)$, por lo tanto:

$$\begin{cases} q_{1_0} = f(\Gamma_{xy}, \zeta_{xy}) = q_{1dm} \pm \zeta_{xy}(k)k_1 \tanh(l_1 \Gamma_{xy}(k)) \\ q_{2_0} = f(\Gamma_{xyz}, \zeta_{xyz}) = q_{2dm} \pm \zeta_{xyz}(k)k_2 \tanh(l_2 \Gamma_{xyz}(k)) \\ q_{3_0} = f(\Gamma_{xyz}, \zeta_{xyz}) = q_{3dm} \pm \zeta_{xyz}(k)k_3 \tanh(l_3 \Gamma_{xyz}(k)) \\ q_{4_0} = f(\Gamma_{xyz}, \zeta_{xyz}) = q_{4dm} \pm \zeta_{xyz}(k)k_4 \tanh(l_4 \Gamma_{xyz}(k)) \\ \vdots \\ q_{na_0} = f(\Gamma_{xyz}, \zeta_{xyz}) = q_{nadm} \pm \zeta_{xyz}(k)k_{na} \tanh(l_{na} \Gamma_{xyz}(k)) \end{cases}$$

donde $\Gamma_{xyz}(k)$ es el producto cruzado resultante de la velocidad y la aceleración en las curvaturas proyectadas en el espacio $\Gamma_{xyz}(k)$ y se define como:

$$\Gamma_{xyz}(k) = \frac{|\dot{\mathbf{r}}(k) \times \ddot{\mathbf{r}}(k)|}{|\dot{\mathbf{r}}(k)|^3}; \text{ with } \mathbf{r}(k) = [\mathbf{h}_x(k) \quad \mathbf{h}_y(k) \quad \mathbf{h}_z(k)]^T \quad (6.75)$$

$\zeta_{xyz} = \text{sign}(\dot{\mathbf{r}}(k) \times \ddot{\mathbf{r}}(k))$ es la dirección de la curvatura en el espacio X, Y, Z ; y k_i, l_i con $i = 1, 2, 3, \dots, na$, son constantes de ganancia que sopesan la incidencia de la curvatura de la trayectoria en función del movimiento de cada una de las articulaciones del brazo robótico.

E. Análisis de estabilidad.

Considerando ya el seguimiento perfecto de la velocidad (es decir, $v_{ref}(t) = v(t)$), la Ecuación (6.71) puede ser sustituida en el modelo cinemático Ecuación (6.64) para obtener la siguiente ecuación de lazo cerrado:

$$\frac{h(k+1) - h(k)}{T_0} = \mathbf{J}(\mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{b} + (\mathbf{I}_n - \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1})\mathbf{v}_0); \quad (6.76)$$

donde $\mathbf{J}\mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1} = \mathbf{I}_m$ la ecuación (6.76) se reduce

$$h(k+1) - h(k) = T_0 \mathbf{I}_m \mathbf{b} + (\mathbf{J}\mathbf{I}_n - \mathbf{I}_m \mathbf{J})\mathbf{v}_0; \quad (6.77)$$

a través de las propiedades de una matriz de identidad se logra

$$h(k) - h(k-1) = T_0 \left(v_d(k) + \frac{W(h_d(k-1) - h(k-1))}{T_0} \right). \quad (6.78)$$

Ahora, definiendo la señal de diferencia Y entre \dot{h}_d y v_{hd} , entonces

$$\begin{aligned} v_d &= \dot{h}_d + \dot{\gamma} \\ h(k) - h(k-1) &= T_0 \left(\dot{h}_d - \dot{\gamma} + \frac{W(h_d(k-1) - h(k-1))}{T_0} \right) \end{aligned}$$

$$h(k) - h(k-1) = T_0 \left(\frac{h_d(k) - h_d(k-1)}{T_0} + \frac{\Delta\gamma}{T_0} + \frac{W(h_d(k-1) - h(k-1))}{T_0} \right)$$

definiendo el error de control como: $\tilde{h}(k-1) = h_d(k-1) - h(k-1)$ en el instante de tiempo $(k-1)$

$$\tilde{h}(k-1) = \tilde{h}(k) + W(\tilde{h}(k-1)) - \Delta\gamma$$

$$\Delta\gamma = \tilde{h}(k) - \tilde{h}(k-1) + W(\tilde{h}(k-1))$$

$$\Delta\gamma = \tilde{h}(k) + \tilde{h}(k-1)(W-1)$$

por el análisis de la transformada Z

$$(1 - z^{-1})\gamma(z) = e(z) + e(z)z^{-1}(W-1)$$

$$(1 - z^{-1})\gamma(z) = e(z)(1 + z^{-1}(W-1))$$

$$\tilde{h}(z) = \frac{1-z^{-1}}{1+z^{-1}(W-1)}\gamma(z); \quad (6.79)$$

donde los polos del sistema son:

$$1 + z^{-1}(W-1) = 0$$

Para que los polos del sistema estén dentro del radio unitario W , debe definirse $0 < W < 1$. Por lo tanto, se concluye que los errores de control $\tilde{h}(k) = 0$ cuando $k \rightarrow \infty$, y tiene estabilidad asintótica.

b. Controlador de compensación dinámica.

El objetivo del controlador de compensación dinámica es compensar la dinámica del robot manipulador móvil, reduciendo así el error de seguimiento de la velocidad. La corrección del error de velocidad genera una optimización energética del sistema robótico, ya que evita movimientos innecesarios para generar grandes descompensaciones.

A. Diseño de la ley de control.

La Figura 6.43 muestra el esquema de control propuesto, el controlador de compensación dinámica recibe como entradas las velocidades deseadas v_c calculadas por el controlador cinemático, y genera referencias de velocidad v_{ref} para el robot manipulador móvil. Por lo tanto, si no hay un seguimiento perfecto de la velocidad, el error de velocidad se define como,

$$\tilde{v}(k) = v_c(k) - v(k) \quad (6.80)$$

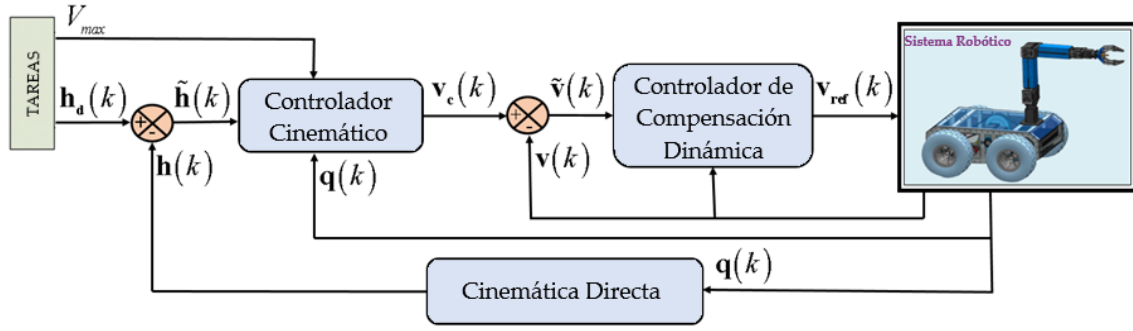


Figura 6.43 Esquema de control propuesto

Este error de velocidad motiva el proceso de compensación dinámica, que se realizará en base a la dinámica inversa del manipulador móvil. Con este objetivo, consideramos el modelo exacto del manipulador móvil sin incluir las perturbaciones.

$$M(q)\dot{v} + C(q, v)v + g(q) = v_{ref} \quad (6.81)$$

Al igual que en (6.63), la aceleración de las velocidades de maniobrabilidad del manipulador móvil puede definirse a través del método de Euler como:

$$\dot{v} = \frac{v(k) - v(k-1)}{T_0} \quad (6.82)$$

Ahora, considerando la aceleración discretizada (6.82) en (6.81) uno tiene

$$v_{ref} = M\left(\frac{v(k) - v(k-1)}{T_0}\right) + Cv + g \quad (6.83)$$

Por la propiedad de Markov:

$$v(k) = v_c(k)$$

Para que los errores tiendan a cero, se define que la velocidad del sistema robótico es

$$v(k) = v_c(k) - W_v(v_c(k-1) - v(k-1))$$

Por lo tanto, se propone la siguiente ley de control:

$$v_{ref}(k) = M\left(\frac{v_c(k) - W_v(v_c(k-1) - v(k-1))}{T_0}\right) + Cv + g \quad (6.84)$$

donde, W_v representa una matriz diagonal constante que acelera los pesos de los errores.

B. Análisis de estabilidad.

Para asegurar la estabilidad del compensador dinámico propuesto, la Ecuación (6.83) con (6.84)

$$M\left(\frac{v(k) - v(k-1)}{T_o}\right) + Cv + g = M\left(\frac{v_c(k) - W_v(v_c(k-1) - v(k-1)) - v(k-1)}{T_o}\right) + Cv + g$$

reduciendo los términos y agrupando, tenemos que el error en el estado actual $\tilde{v}(k)$ depende sólo del error anterior multiplicado por una matriz de ganancia W_v , es decir,

$$\begin{aligned} v_c(k) &= v(k) + W_v(\tilde{v}(k-1)) \\ \tilde{v}(k) &= W_v(\tilde{v}(k-1)) \end{aligned} \quad (6.85)$$

Considerando que: $\tilde{v}(k) = [\tilde{v}_u(k) \quad \tilde{v}_\omega(k) \quad \tilde{v}_{\dot{q}_1}(k) \quad \dots \quad \tilde{v}_{\dot{q}_{na}}(k)]^T$;

$$\tilde{v}(k-1) = [\tilde{v}_u(k-1) \quad \tilde{v}_\omega(k-1);$$

$$\tilde{v}_{\dot{q}_1}(k-1) \quad \dots \quad \tilde{v}_{\dot{q}_{na}}(k-1)]^T \text{ and } W_v = \text{diag}(w_u, w_\omega, w_{\dot{q}_1}, \dots, w_{\dot{q}_{na}});$$

la ecuación (6.85) puede ser representada por:

$$\begin{bmatrix} \tilde{v}_u(k) \\ \tilde{v}_\omega(k) \\ \tilde{v}_{\dot{q}_1}(k) \\ \vdots \\ \tilde{v}_{\dot{q}_{na}}(k) \end{bmatrix} = \begin{bmatrix} w_u \tilde{v}_u(k-1) \\ w_\omega \tilde{v}_\omega(k-1) \\ w_{\dot{q}_1} \tilde{v}_{\dot{q}_1}(k-1) \\ \vdots \\ w_{\dot{q}_{na}} \tilde{v}_{\dot{q}_{na}}(k-1) \end{bmatrix}$$

Ahora, si analizamos los errores cuando $k = 1, 2, 3, \dots, n$ y considerando $i = u, \omega, \dot{q}_1, \dot{q}_2, \dots, \dot{q}_{nq}$, se tiene:

$$\begin{aligned} \tilde{v}_i(k) &= w_i \tilde{v}_i(k-1) \\ \tilde{v}_i(k+1) &= w_i \tilde{v}_i(k) = w_i^2 \tilde{v}_i(k-1) \\ \tilde{v}_i(k+2) &= w_i \tilde{v}_i(k+1) = w_i^3 \tilde{v}_i(k-1) \\ &\vdots \\ \tilde{v}_i(k+n) &= w_i \tilde{v}_i(k+n-1) = w_i^n \tilde{v}_i(k-1) \end{aligned}$$

Por lo tanto, si $0 < w_i < 1$, se puede concluir que el error de velocidad del sistema robótico del manipulador móvil tiende asintóticamente a cero cuando $n \rightarrow \infty$.

Finalmente, el esquema de control propuesto para resolver el problema de control de movimiento del robot manipulador móvil se muestra en las Figuras 6.40 y 6.43, donde el diseño del controlador se basó en dos subsistemas en cascada: i) Controlador cinemático: para este controlador se propone (6.71), que resuelve el problema de control de movimiento del sistema robótico considerando la posición y velocidad deseadas del efector final del robot manipulador móvil para la ejecución de la tarea deseada; ii) Controlador de compensación dinámica: para compensar la dinámica del manipulador móvil, reduciendo así el error de seguimiento de la velocidad, se propone la ley de control (6.84), donde $v_{ref}(k)$ representa la salida del controlador

de compensación dinámica, es decir, $v_{ref}(k)$ es el vector de las velocidades de entrada al sistema robótico manipulador móvil.

6.3.4 Resultados experimentales.

Los resultados se presentan en dos partes: la construcción mecánica-electrónica y las pruebas experimentales desarrolladas para validar los controladores propuestos. La construcción presenta un breve resumen del prototipo creado para ejecutar los experimentos, subdividido en construcción mecánica y electrónica, mientras que las pruebas experimentales se apoyan en diversos gráficos que muestran la eficacia de los controladores propuestos, además de validar la optimización de la energía.

a. Plataforma mecánica y electrónica.

Como se mencionó en el capítulo III del presente trabajo, en esta aplicación se usa las construcciones mecánicas desarrolladas, específicamente la plataforma móvil uniclo y el brazo robótico Figuras 3.4 y 3.30, estos generan movimientos en el manipulador móvil, con dos motores laterales que dan movimiento a sus ruedas laterales tal como se observa en la Figura 6.44, que además contiene internamente todos los elementos asociados a su control electrónico y sistema de comunicaciones, instalados de forma estratégica que permitan recibir de forma normal a la colocación del brazo robótico construido, que se muestra en la Figura 6.45.

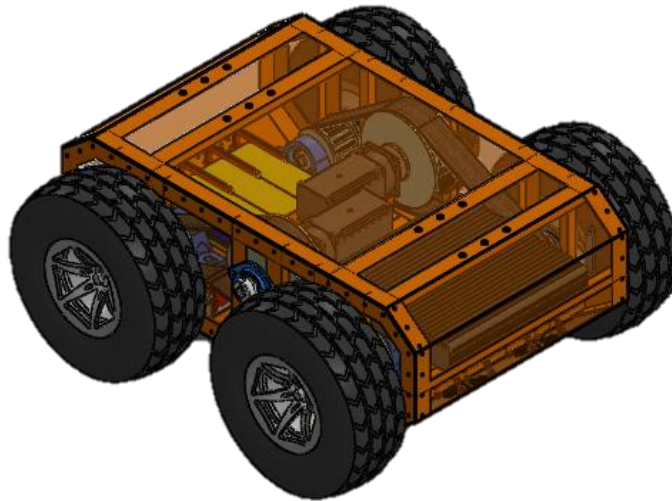


Figura 6.44 Configuración del robot móvil base del manipulador móvil.

Para el brazo robótico, se consideraron todos los mecanismos para poder producir los movimientos necesarios para el desarrollo de las tareas y además que tuviera dimensiones tales que se monte de forma normal sobre la plataforma móvil.

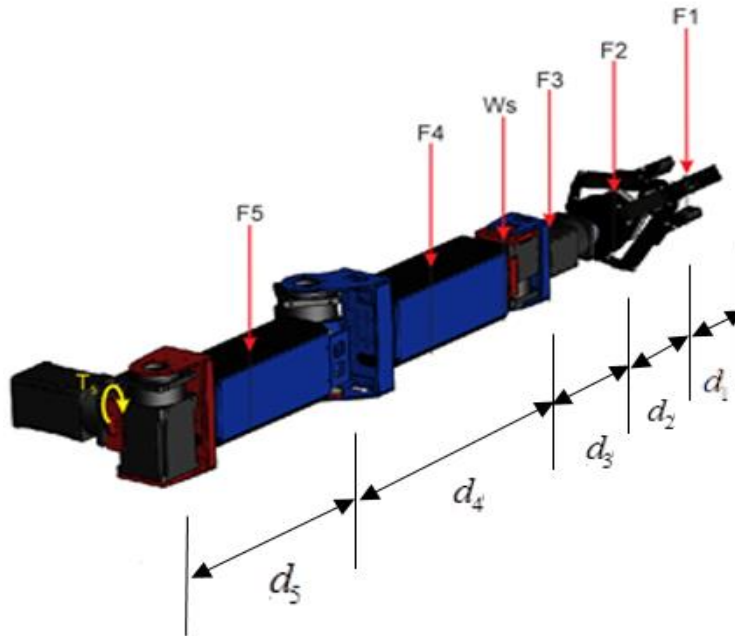


Figura 6.45 Configuración del brazo robótico elemento del manipulador móvil

Dado que la construcción de cada parte mecánica está diseñada para encajar, el resultado del análisis del centro de masas y la distribución apropiada del peso permite que la plataforma y el brazo se unan en un único sistema robótico, como se muestra en la Figura 6.46.

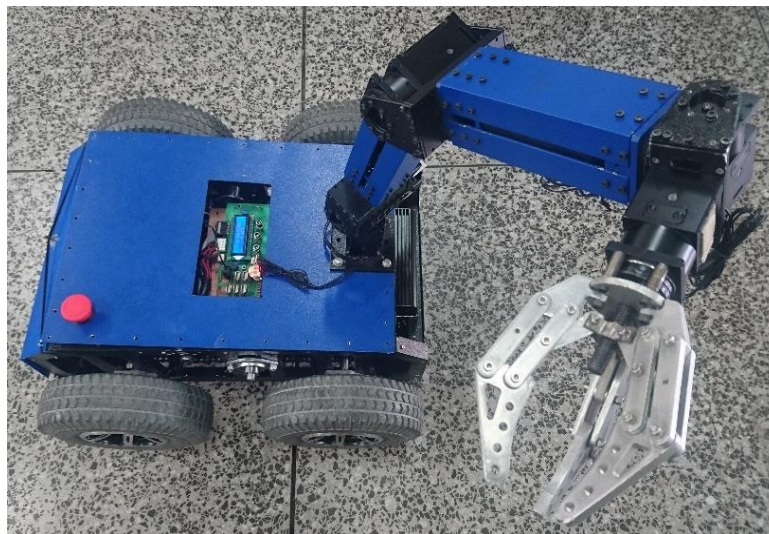


Figura 6.46. Manipulador móvil DUT

Los componentes electrónicos del robot manipulador móvil están diseñados para interconectar todos los equipos y elementos de control, alimentación y suministro de energía. El diseño consta de ocho motores Dynamixel, una computadora, un tablero de control electrónico, elementos de prevención de sobrecorriente, un extensor periférico y un banco de baterías de polímero de litio

(LiPo). La distribución de los elementos se muestra en la Figura 6.47, que además denota la comunicación entre ellos.

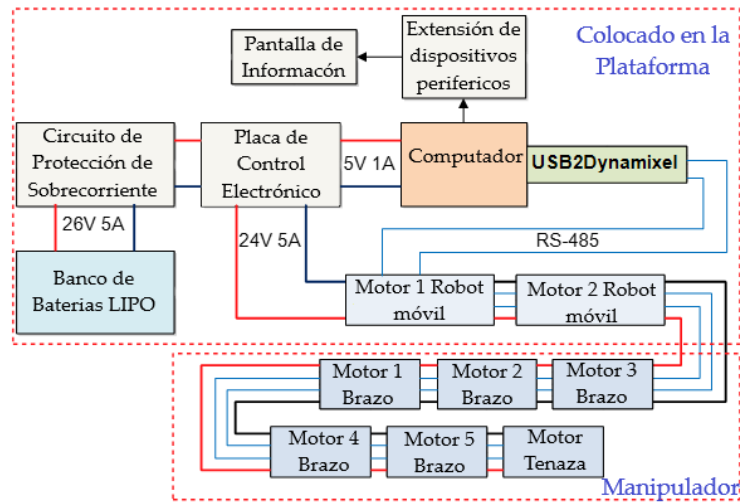


Figura 6.47. Sistema electrónico del manipulador móvil DUT

- i) Los motores de Dynamixel, a diferencia de los motores con control de frecuencia, pueden modificar sus características mediante una trama de comunicación en serie. La comunicación con la computadora se ve facilitada por la inclusión de un dispositivo USB2Dynamixel, que entrama la información para optimizar la comunicación y evitar la saturación de la red en serie. La velocidad de comunicación entre la computadora y toda la red se optimiza con la creación de librerías que permiten invocar funciones programadas en lenguaje C a través de archivos Mex. Con la orientación a este trabajo, la información de corriente, velocidad, posición y par se adquiere por la trama serial. La información de corriente permite conocer la magnitud exacta de la potencia consumida por el sistema robótico, por lo que no es necesario incluir equipos de medición adicionales a la información proporcionada por los actuadores. ii) La computadora tiene recursos suficientes para ejecutar programas de procesamiento de alto nivel. En la computadora, los cálculos necesarios para los controladores se realizan en programas matemáticos. (iii) El tablero de control electrónico regula y distribuye los voltajes de potencia a todo el motor y el sistema informático. La placa incorpora elementos de medición de corriente y tensión para la emisión de señales de aviso en posibles descargas o desconexiones de elementos. Además, la tarjeta de control electrónico incorpora un indicador de estado de carga y conexión. iv) Se incluye un extensor de puertos periféricos para proporcionar comunicación entre los dispositivos externos (cámaras, memorias de SD o flash, etc.) y la computadora interna. Por último, v) un banco de baterías LiPo proporciona la

energía necesaria para el sistema, que suministra hasta 15 A/h con un soporte de corriente de pico de hasta 25 A.

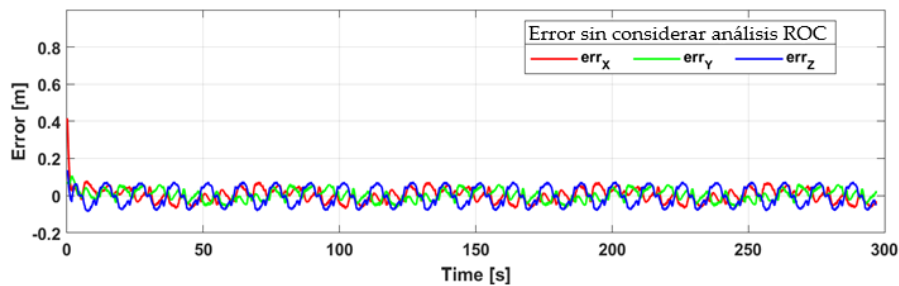
b. Evidencia experimental.

Para evaluar el rendimiento de la propuesta de ahorro de energía, se desarrollaron tres experimentos diferenciados por la forma geométrica del camino a seguir: silla, flor y función senoidal. Cada uno de los experimentos se ejecutó con diferentes propósitos: medir el consumo de corriente absorbida, el número de metros recorridos o el tiempo de ejecución. Hay que aclarar que los experimentos se consideran ejecuciones en diferentes caminos, mientras que las pruebas se consideran ejecuciones en el mismo camino, pero con el uso de diferentes controladores (considerando o no el análisis del radio de curvatura, que se denomina ROC).

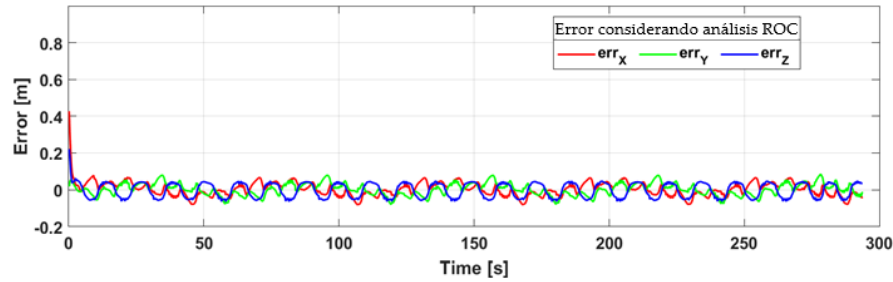
El primer experimento considera un camino deseado descrito por los medios $h_d = [x_d, y_d, z_d]$, donde: $x_d = 1.5 \cos(0.1t) + 1.75$, $y_d = 2.5 \sin(0.1t) + 2.75$, $z_d = 0.15 \sin(0.2t) + 0.6$.

La selección de este camino en particular fue porque busca demostrar directamente la acción de la reducción de la velocidad al encontrar una curva. Para demostrar el ahorro de energía a través de resultados contrastados, se propusieron dos tipos de ejecuciones, ambas considerando el controlador basado en Lyapunov con redundancia en el brazo robótico, pero sólo una con la inclusión de la saturación de velocidad por el análisis del radio de la curvatura. El seguimiento de la trayectoria para ambas configuraciones se evaluó sobre la misma distancia recorrida, en este caso, 100 metros.

El resultado considerando sólo el control con redundancia presentó errores importantes en cada uno de los ejes (Figura 6.48); y, por lo tanto, un mayor error total comparado con el resultado considerando el análisis de radio de curvatura, como se muestra en la Figura 6.49. Destacar que, a partir de ahora, el ROC se definirá como el radio de curvatura.



(a)



(b)

Figura 6.48 Errores en cada uno de los ejes (a) sin considerar el análisis ROC (radio de curvatura), (b) considerando el análisis ROC.

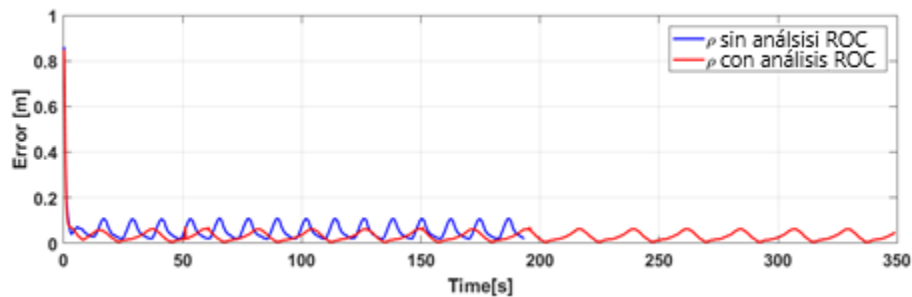
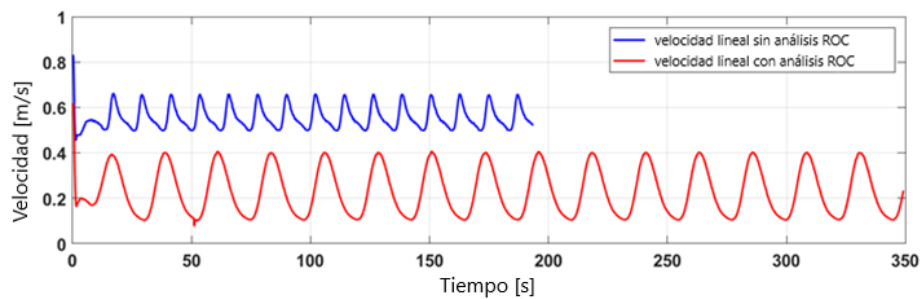


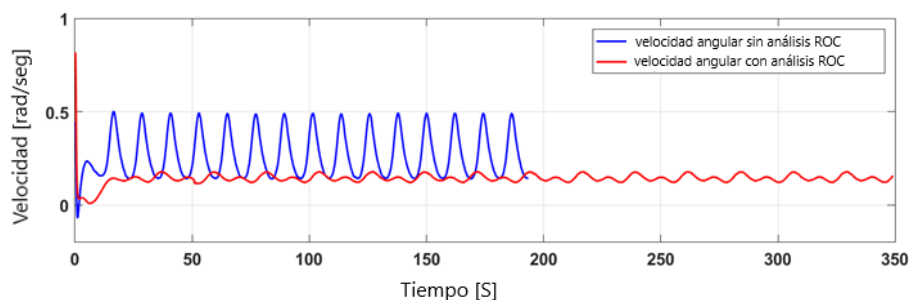
Figura 6.49 Errores de control cuadrático.

Esta reducción de los errores en cada uno de los ejes se debió a que la plataforma móvil reduce las velocidades de control al acercarse a una curva, como se muestra en la comparación de las velocidades lineales y angulares de la Figura 6.50.

Además de reducir la velocidad de ejecución frente a las curvas, el algoritmo propuesto da más peso a las acciones correctivas del brazo robótico, reduciendo significativamente los movimientos de la plataforma móvil y, por tanto, el consumo de energía. Las acciones de control aplicadas al brazo robótico para ambas pruebas (con y sin consideración del análisis ROC) se muestran en la Figura 6.50.



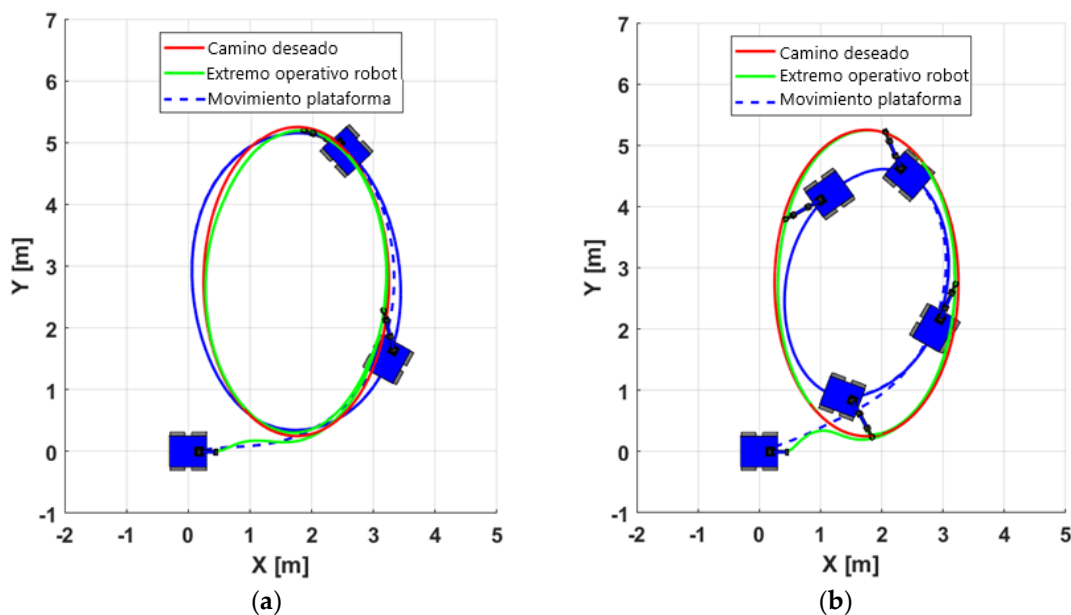
(a)



(b)

Figura 6.50 Acciones de control de la plataforma para la ejecución del seguimiento sin consideración del análisis ROC y con consideración del análisis ROC. a) brazo robótico, b) robot móvil.

El movimiento estroboscópico del sistema robótico permite determinar la ejecución resultante aplicando ambos controladores. A través de una vista superior, la Figura 6.51 muestra la comparación entre el movimiento estroboscópico del sistema robótico sin tener en cuenta el análisis del radio de curvatura y con la consideración del análisis del radio de curvatura, diferenciando la trayectoria de la plataforma y los movimientos ejecutados por el brazo robótico. La Figura 6.52 también presenta una vista estroboscópica de los movimientos ejecutados por el robot en el espacio.



(a)

(b)

Figura 6.51 Vista superior del movimiento estroboscópico de la trayectoria siguiendo a) sin considerar el análisis del radio de curvatura y b) con consideración del análisis del radio de curvatura.

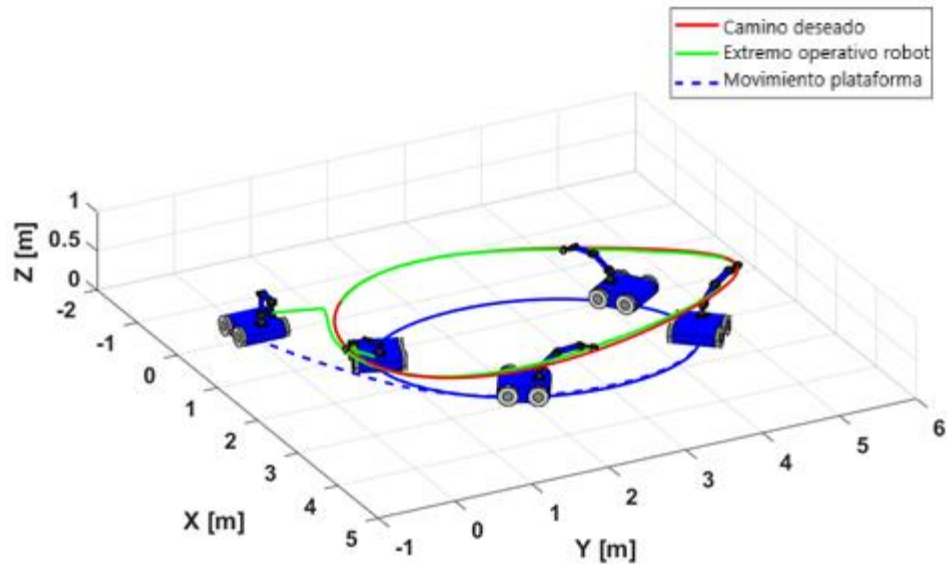


Figura 6.52 Movimiento estroboscópico en el espacio del sistema robótico para el primer experimento que considera el análisis ROC.

El consumo de energía de los actuadores del sistema robótico fue medido a través de una de las propiedades de la red de motores. La medición de la corriente consumida se muestra en la Figura 6.53, donde se aprecia el contraste entre el consumo de energía considerando el radio de análisis de la curvatura y sin su consideración. Como puede verse, dada la reducción de la velocidad del sistema robótico, el tiempo de ejecución se incrementó para cumplir con la tarea asignada. Sin embargo, después de ejecutar completamente la tarea, el controlador propuesto resultó en un ahorro de energía del 8,76%, como se muestra en la Tabla 6.3.

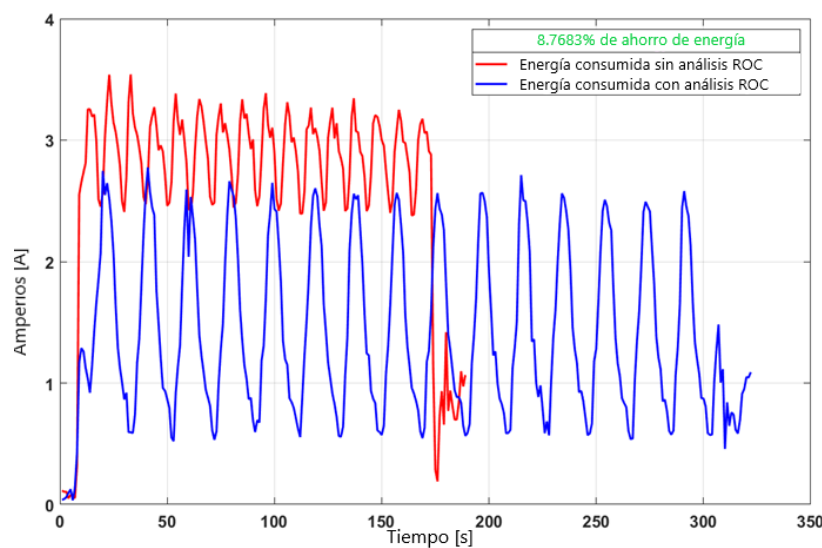


Figura 6.53 Medidas del consumo de corriente

Tabla 6.3 muestra el 8,76% de ahorro de energía causado por el controlador.

Parámetro	SIN Análisis ROC	CON Análisis ROC
Distancia [m]	100	100
Vmax [m/s]	0.5	0.5
Tiempo total transcurrido [s]	190	325
Promedio Corriente [A]	2.613	1.399
Voltaje [V]	29	29
Energía consumida [J]	493.273	450.021
Máximo pico [A]	3.539	2.775
Mínimo pico [A]	2.409	0.521
Promedio Potencia [W]	75.777	40.571

El camino considerado en el segundo experimento está descrito por $h_d = [x_d, y_d, z_d]$ para valores de: $x_d = 1.5 \cos(0.1t) + 1.75$, $y_d = 2.5 \sin(0.1t) + 2.75$ and $z_d = 0.15 \sin(0.2t) + 0.6$.

A diferencia del primer experimento, la trayectoria establecida indicaba la eficiencia del controlador estableciendo una trayectoria que contenía curvas pronunciadas en todos los componentes (Figura 6.54). Este experimento proponía determinar la distancia total recorrida considerando la misma cantidad de energía consumida con y sin análisis de radio de curvatura, asumiendo así la similitud entre los errores de control de ambas pruebas. Esta consideración requería encontrar las velocidades máximas para el controlador con y sin análisis de radio de curvatura que satisficieran la igualdad de errores, a fin de hacer más equitativa la comparación entre los dos consumos de energía. En la Figura 6.55 se muestran los errores en cada uno de los ejes, mientras que en la Figura 6.56 se muestran los errores cuadráticos medios de ambas pruebas con velocidades máximas de 0,25 m/s en el controlador sin análisis ROC y con una velocidad máxima de 0,45 m/s para el controlador con análisis ROC.

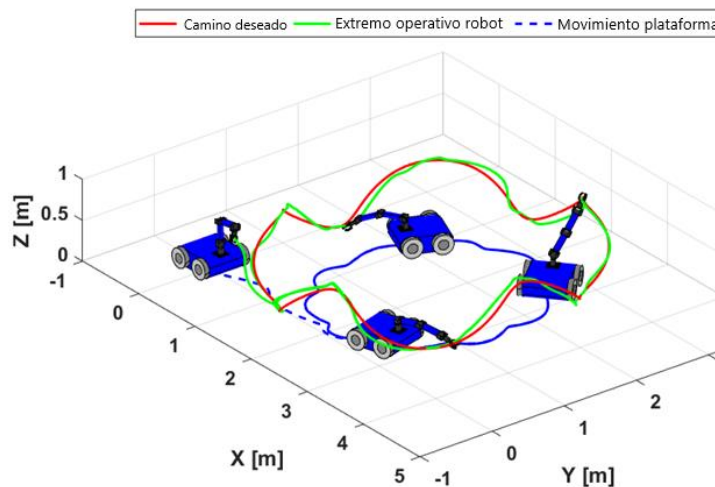
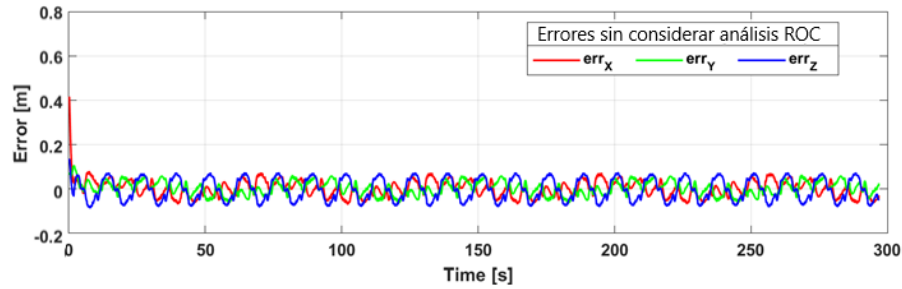
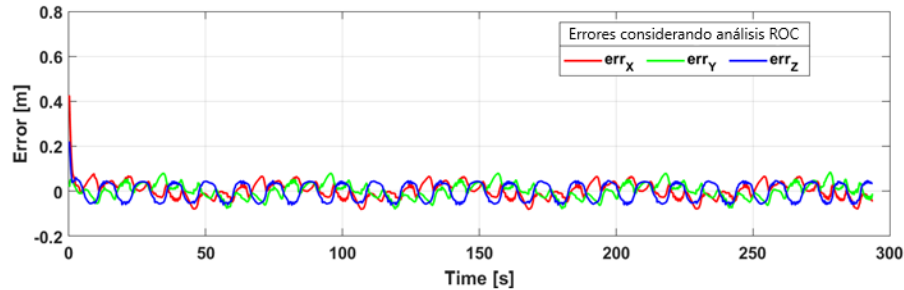


Figura 6.54 Trayectoria considerada para el segundo experimento



(a)



(b)

Figura 6.55 Errores en cada uno de los ejes del segundo experimento: a) sin considerar el análisis ROC y b) considerando el análisis ROC.

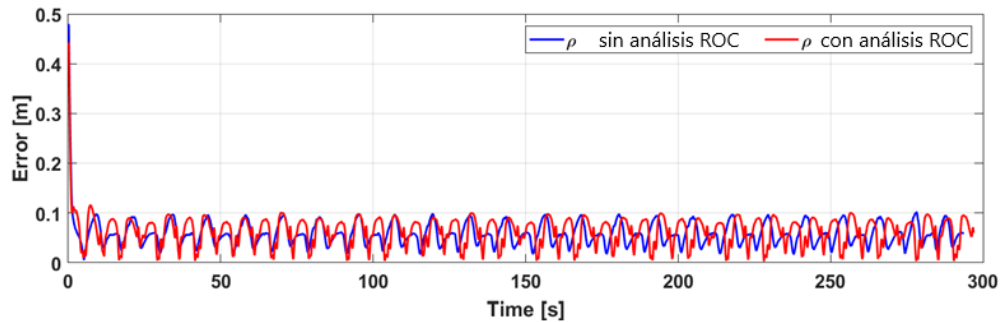


Figura 6.56 Errores cuadráticos totales en el segundo experimento

La Figura 6.57 muestra la comparación desde una vista superior del recorrido de la plataforma sin la inclusión del análisis de radio de curvatura y con la inclusión del mismo, donde el desplazamiento en metros del segundo es menor en comparación con la ejecución del primero. Este resultado se da porque, además de la inclusión del análisis de radio de curvatura, los pesos de redundancia en el brazo robótico fueron mayores que los de la plataforma móvil, reduciendo considerablemente los movimientos de los motores que permitían el desplazamiento en el plano.

La corriente consumida por ambas pruebas a lo largo de la duración del recorrido (a una distancia total de 70 metros) se muestra en la Figura 6.58; el consumo total de energía de la prueba sin considerar el análisis de curvatura del radio fue de 302.464 julios, mientras que con la consideración del análisis ROC fue de 208.102 julios. A partir de estos resultados, se puede

determinar que el consumo del sistema robótico cuando no se consideró el análisis de curvatura fue de 4.321 J/m (0,23 m/J), mientras que el consumo cuando se consideró el análisis ROC fue de 4.001 J/m (0,25 m/J). Suponiendo un consumo de 500 julios como límite máximo para ambas pruebas, la distancia recorrida por el controlador sin análisis de curvatura fue de 115 metros, mientras que para el controlador con optimización fue de 125 metros.

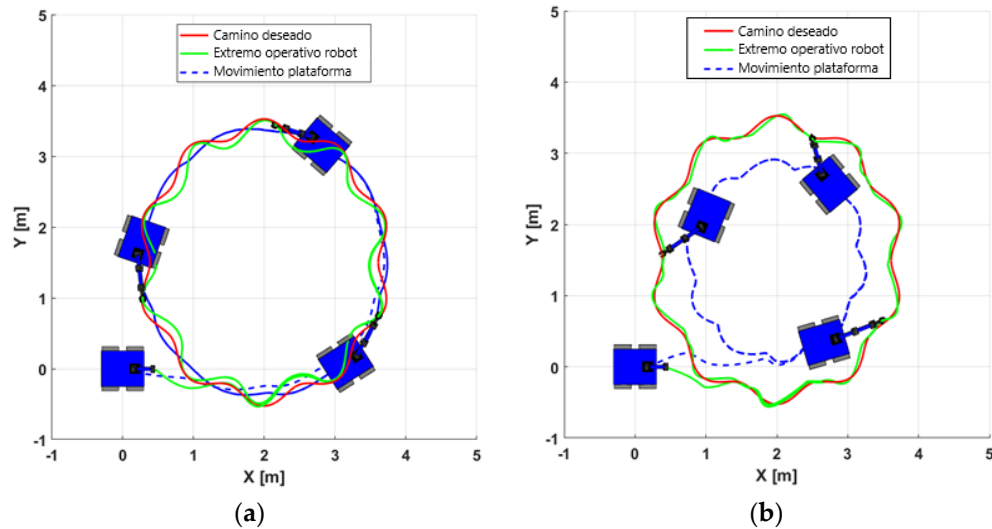


Figura 6.57 Vista superior del recorrido del sistema robótico sin considerar el análisis de curvatura y con consideración del análisis de curvatura. a) sin análisis ROC, b) con análisis ROC.

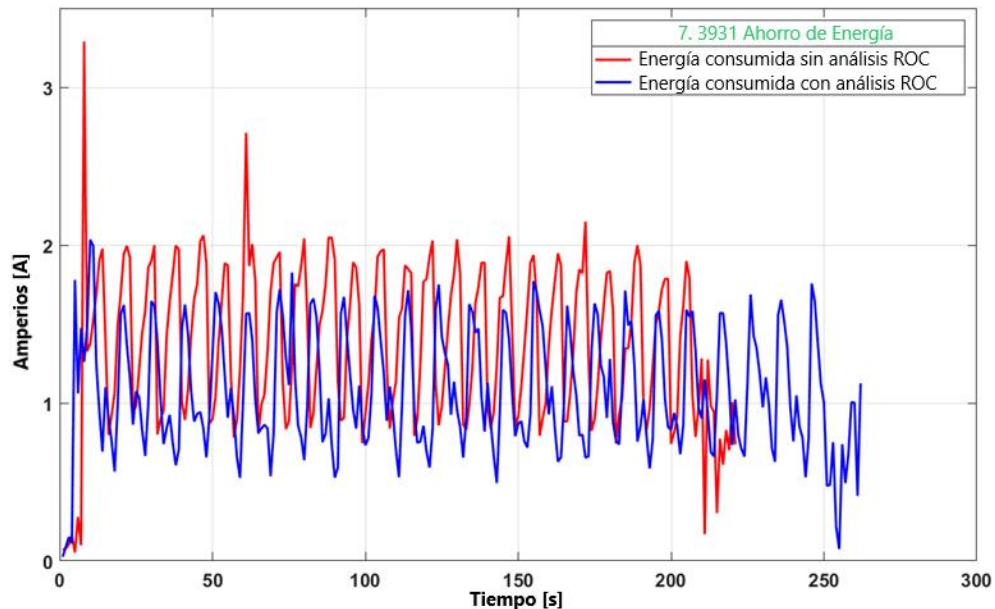


Figura 6.58 Consumo de energía en ambas pruebas: sin análisis ROC y con análisis ROC

Como se muestra en las velocidades de control (Figura 6.59), la consideración del análisis del radio de curvatura hizo que los valores de entrada a la plataforma robótica se redujeran en presencia de una curva, ampliando el tiempo de ejecución, pero optimizando al mismo tiempo el consumo de energía.

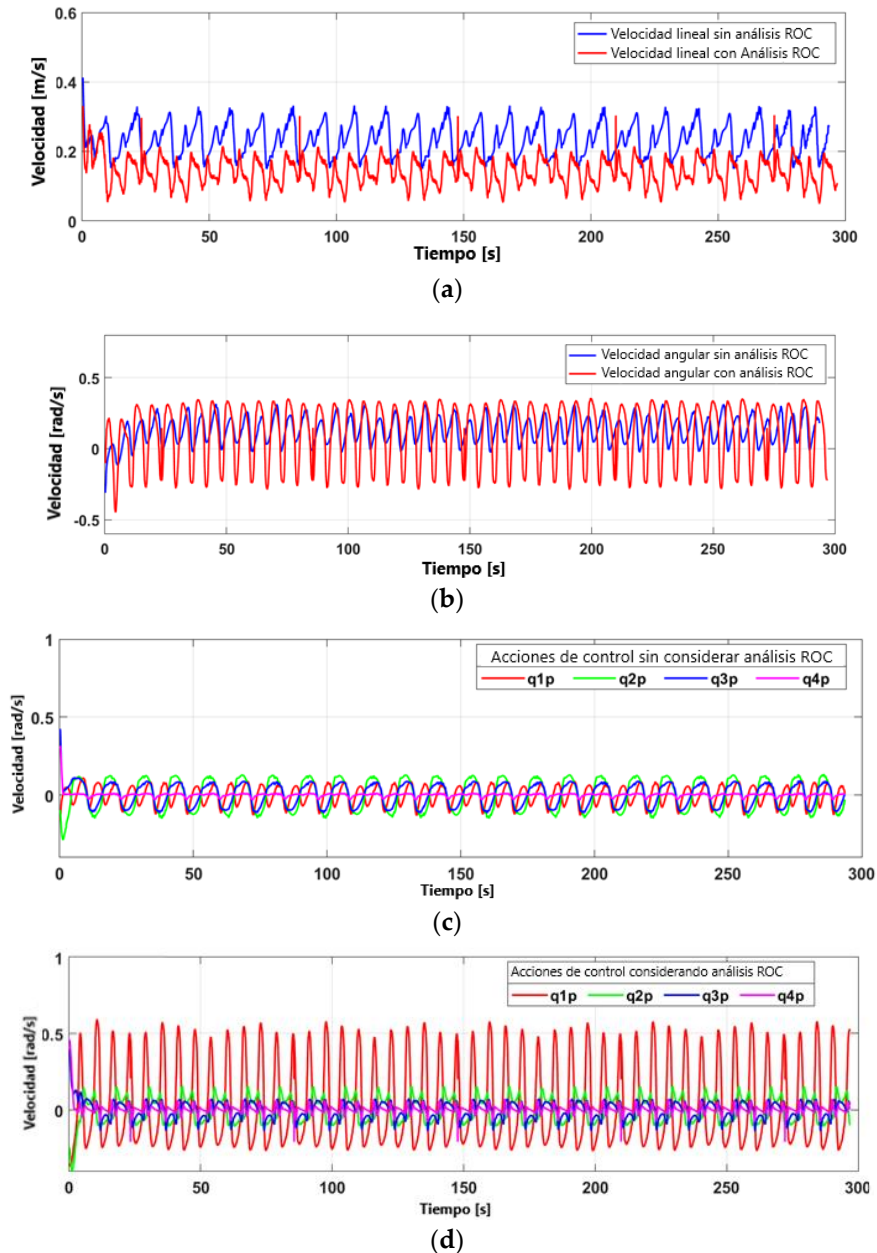


Figura 6.59 Control de las velocidades de todo el sistema robótico. (a) velocidad lineal reducida, (b) velocidad angular reducida, (c) acciones de control lineales reducidas, (d) acciones de control angulares reducidas.

Por último, el tercer experimento presenta la ejecución de cuatro pruebas considerando un controlador basado en Lyapunov para el seguimiento de caminos, pero con características

diferentes: *i)* control del sistema robótico sin consideración de la manipulabilidad, *ii)* control de la plataforma móvil mediante el análisis del radio de curvatura pero sin consideración de la manipulabilidad, *iii)* control del sistema robótico considerando la manipulabilidad, *iv)* control del manipulador móvil considerando la manipulabilidad y aplicación de la saturación de velocidad para todo el sistema robótico. Para comparar todas las pruebas, la trayectoria seleccionada se muestra en la Figura 6.60, además de mostrar los movimientos estroboscópicos ejecutados por el sistema robótico teniendo en cuenta la manipulabilidad y la aplicación de la saturación de velocidad dada por el análisis del radio de curvatura en la Figura 6.61.

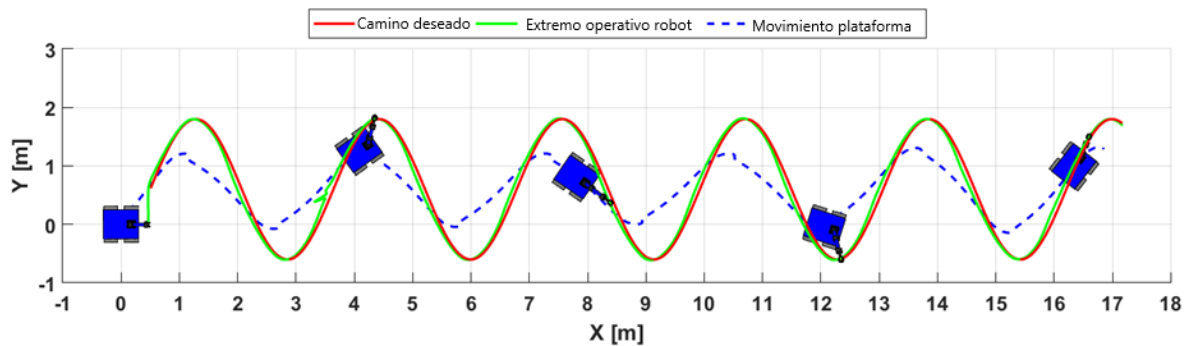


Figura 6.60 Camino seleccionado para el tercer experimento considerando el análisis ROC y la manipulabilidad.

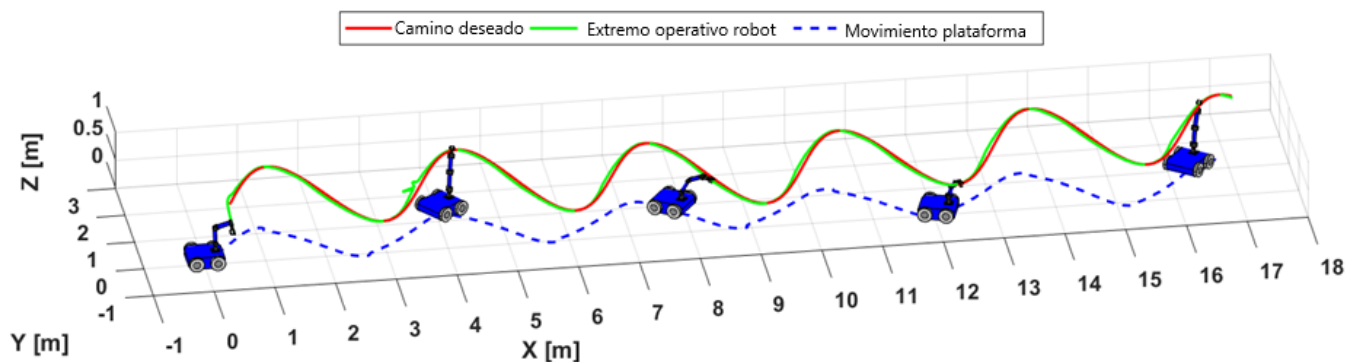


Figura 6.61 Movimiento estroboscópico de la ejecución del tercer experimento considerando el análisis ROC y la manipulabilidad.

El consumo de energía en todos los casos fue mayor en relación con el gasto de energía del controlador propuesto para la optimización de la energía. Esto se verifica a través del análisis de la representación gráfica del consumo de energía que se muestra en la Figura 6.62, así como por la Tabla 6.4. En la Tabla 6.4 se resumen los resultados obtenidos al someter el sistema robótico a las distintas configuraciones de los controladores, comparando el ahorro energético de cada uno de ellos con el ahorro obtenido por el controlador propuesto.

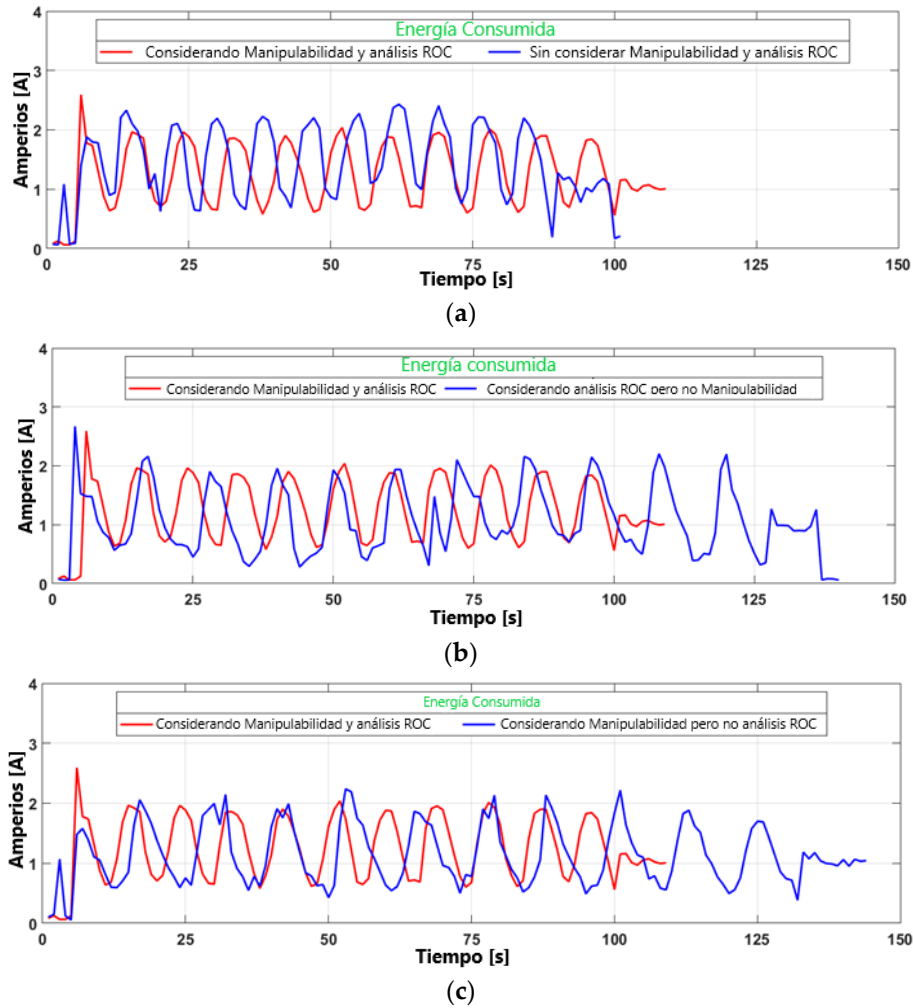


Figura 6.62 Consumo de energía en cada una de las pruebas vs. controlador considerando el análisis ROC (rojo). (a) controlador propuesto 1, (b) controlador propuesto 2, (c) controlador propuesto 3.

Tabla 6.3 Resultados obtenidas para las distintas configuraciones.

Configuración	Primera	Segunda	Tercera	Cuarta
Distancia [m]	32.74	32.42	31.86	34.1
Corriente promedio [A]	1.43	1.07	1.16	1.13
Potencia promedio [W]	42.07	31.44	34.37	33.33
Energía [J]	4245.36	4400	4224.48	3976.05
Voltaje [V]	29.4	29.4	29.4	29.4
Error acumulado	37.77	36.48	28.79	27.65
Energía ahorrada [%]	-6.34	-9.63	-5.88	0%

Observación 7: Primera configuración: Sin manipulabilidad ni análisis ROC; 2ª Configuración: Considerando el análisis ROC pero no la manipulabilidad; 3ª Configuración: Considerando la

manipulabilidad pero no el análisis ROC; 4ª Configuración: Considerando la manipulabilidad y el análisis ROC.

La propuesta integra en un solo análisis matemático los dos sistemas (brazo robótico y robot móvil unicycle) de tal manera que aprovecha sus características eficientes en la búsqueda de la reducción del consumo de energía en las tareas que realiza. Se han añadido sofisticados elementos matemáticos para evaluar su acción en el controlador, y se ha demostrado en pruebas experimentales por una reducción del 8,76% del consumo de energía en las tareas.

Esto se ha basado en un complejo diseño mecánico y electrónico que integra sistemas de control y comunicaciones inteligentes y una plataforma mecánica eficiente que facilita la máxima versatilidad en los movimientos, y por lo tanto no introduce pérdidas de energía debido al diseño mecánico.

Capítulo VII

7 CONCLUSIONES Y TRABAJO FUTURO.

En este trabajo de tesis doctoral se han realizado muchas actividades, persiguiendo el objetivo central y los objetivos particulares que se definieron en el proyecto original, pero así mismo, con el paso de los años, se fueron descubriendo otros temas que contribuían a conseguir estos objetivos a través de metas y objetivos más pequeños, que renovaban el espíritu de la búsqueda de las mejores soluciones y definían nuevos caminos, dependientes del progreso de la tecnología y el desarrollo de la ciencia de todos los días.

Como podemos leer en el proyecto el objetivo central fue la construcción de una nueva plataforma robótica móvil que pudiera ejecutar tareas relacionadas a la búsqueda, a su propia movilidad y sustentabilidad y algunas funciones más, como el ahorro de energía en las tareas, la colaboración, su supervisión, etc. Para finalizarlo se quiere puntualizar los aspectos principales de la consecución de estos objetivos y además los muchos trabajos que a partir de las soluciones implementadas se pueden realizar en el futuro.

7.1 Conclusiones.

1. En este trabajo, conforme se plantea en el proyecto, se ha desarrollado una enorme cantidad de hardware y software que pueda producir las aplicaciones necesarias para comprobar los preceptos que se habían planteado, es decir la autonomía de control y operación de los robots, con el error de localización casi en sus límites a cero, el ahorro de energía y la capacidad innata de colaboración, además de la posibilidad de monitorear su tarea individual o cooperativa por un sistema de observación, control o supervisión.
2. En cuanto al hardware desarrollado, se ha emprendido un sistema completo de diseño mecánico, que permita de forma muy técnica y científica implementar en su control algoritmos sumamente inteligentes a los cuales deba responder de forma efectiva y eficiente, con la finalidad de que sus características particulares no incluyan errores sistemáticos o no sistemáticos que puedan ser incorregibles por software.
3. En lo que tiene que ver con el software; se ha desarrollado programas de características muy avanzadas, que permitan en primer lugar el control autónomo del movimiento y desplazamiento de las plataformas móviles de cualquier naturaleza (se centró básicamente en dos tipos de robots móviles: los unicycle y los omnidireccionales); algoritmos de control de cada tarea a desarrollarse de forma independiente a la tarea del control del movimiento, y finalmente el software de supervisión que desarrolla un ambiente de realidad virtual en donde se puede realizar tareas de esta naturaleza con el/los robots intervinientes.

4. A pesar de que, el planteamiento del tema y los objetivos particulares de esta tesis, podía sugerir el desarrollo de una única y novedosa plataforma robótica que al ser multisensorial se pudiera usar para múltiples aplicaciones (o todas las que puedan existir); NO existe una plataforma única, que sea la panacea en robótica móvil ni colaborativa, pues hemos comprobado que a pesar de que, en la actualidad, se pueden introducir todos los sensores fabricados en el mercado para cualquier aplicación, en los robots móviles (más aún si están configurados con protocolos o sistemas estándar como I2C, serial, etc.), cada aplicación requiere uno o un conjunto de sensores particulares para realizar sus tareas específicas, y el hecho de que se incluyan más sensores de los necesarios redundaría directamente en el costo computacional, consumo de potencia, peso, capacidad y algunos otros aspectos que se pueden considerar adversos en las aplicaciones.
5. Se implementó un sistema que con una combinación de hardware y software, asegura la reducción de errores de odometría al mínimo, este sistema emplea de forma primaria un control de estabilidad del movimiento empleando un controlador que desarrolla un algoritmo de control tipo PID para cada rueda motriz del robot (este sistema es empleado luego, en todas las aplicaciones) y luego combina esto con la introducción de otro controlador que actúa mediante visión artificial y controla el error producido por odometría y efectos externos al robot.
6. Se realizó un estudio de la red inalámbrica más conveniente para aplicaciones de robótica colaborativa, los parámetros de elección se determinaron en cuanto a velocidad, distancia, complejidad, costo, disponibilidad, funcionalidad, latencia y otros más, que hacen que los robots de características heterogéneas operen adecuadamente en un ambiente de colaboración, y además pueda extenderse esta red a más nodos con diferentes operaciones, como sistemas de teleoperación y supervisión.
7. Se desarrolló un sistema de colaboración entre robots móviles de características heterogéneas, que implementa algoritmos matemáticos sofisticados para el seguimiento de caminos, búsqueda de las posiciones de los demás robots y mantenimiento de una trayectoria como tareas básicas del conjunto robótico; como característica principal, este sistema, no considera dentro del controlador principal las características cinemáticas de los robots que interactúen entre sí, sino más bien permite que robots de distintas características se integren a él; dejando estas características para desarrollarlas en los controladores de movimiento empleados en cada robot.
8. Se desarrolló un sistema que comprobó la reducción del consumo de energía de robots móviles tipo manipulador móvil, usando las velocidades combinadas de las tareas asignadas al robot móvil y al brazo robótico, en este sistema se tomó como un conjunto a las tareas que desarrollan ambos robots simultáneamente y empleo un novedoso sistema de control llamado análisis del radio de curvatura en el que, se define la tarea de mayor austeridad

energética en tareas simultáneas del móvil y el brazo, eligiendo quien de forma particular consume menos energía para su consecución.

9. Se desarrolló un sistema de supervisión basado en software de juegos de computador, combinado con el software de control, que usa API's de sistema operativo para comunicar entre las aplicaciones de control y supervisión al ambiente; y desarrollar de esta forma sistemas de monitoreo, control (teleoperación) y supervisión en tres dimensiones y orientados al uso de la realidad virtual.
10. Como mecanismo de difusión de resultados se realizaron tres publicaciones internacionales en revistas y congresos de alto impacto, que fueron publicadas en el año pasado.

7.2 Trabajos futuros.

Con la experiencia obtenida en esta tesis, tanto en el desarrollo del hardware como del software de aplicación, surgen nuevos retos que se pueden enfrentar con la base de los sistemas desarrollados; al final, se ha desarrollado un gran sistema que puede aplicarse en otros ámbitos y servicios, se considera que, a partir de los desarrollos obtenidos, se puede seguir desarrollando también:

1. En el ámbito de la robótica y para poder emplear una generalidad de los algoritmos desarrollados, se puede desarrollar robots aéreos, acuáticos y subacuáticos, que puedan ser controlados con las mismas técnicas de movimiento planteadas, incluyendo en ellas la carga que cada plataforma deberá enfrentar es decir el ambiente sobre el que se va a desenvolver, es decir: aire, sobre y bajo el agua; se deberá determinar el sensor o grupo de sensores que ha de implementarse así como también sistemas que impidan la invasión del ambiente en el entorno eléctrico.
2. Comprobar la eficacia de distribuir tareas de control, en tareas de control más simples, pequeñas y segmentadas, que sean desarrolladas por placas o sistemas embebidos en casos particulares, con el fin de tener en la totalidad o en lo parcial, un reducido consumo de energía y el costo computacional en sistemas operativos de tiempo real.
3. Extender la aplicación de corrección de errores de odometría en ambientes no estructurados, en los cuales el robot se desplace por superficies más extensas que el modelo de acción implementado, y la colocación de la cámara web, que, en lugar de estar estática sea portada por un dron, de forma que en este sistema el robot incluso, a más de corregir sus errores de odometría, pueda realizar SLAM del ambiente no estructurado circundante.
4. Que pueda también comprobarse la eficacia del algoritmo de colaboración propuesto, en sistemas aéreos UAV's (fuera de la tesis en el grupo de investigación se está trabajando ya en eso) para demostrar su operación en ambientes con perturbaciones muy agresivas como vientos y brisas para determinar la robustez del algoritmo de compensación y poder, en un

momento determinado integrar ambos sistemas: terrestre y aéreo para aplicaciones de uso militar especialmente. Extender este sistema también a conjuntos de robots subacuáticos.

5. Aplicar el conocimiento y experiencia en el diseño de un sistema de colaboración que mantenga la formación y tareas primarias y secundarias, para robots manipuladores móviles de tal manera que se siga la trayectoria con o sin la modificación de la formación desde los efectores finales de cada manipulador móvil, y extender nuevamente a sistemas híbridos aire tierra, incorporando en él a los UAV's.
6. Combinar lo realizado en el sistema de supervisión, con dispositivos modernos de realidad virtual y realidad aumentada, de tal forma que el operador humano que administre la supervisión, no se circunscriba únicamente al monitor de una computadora, sino extienda esta funcionalidad a su propio cuerpo, de esta forma comprobar la rapidez de la respuesta en el sistema y prevenir situaciones imprevistas gracias a la presencia humana en la supervisión.
7. Extender la cobertura de la red inalámbrica a ambientes de operación de robótica móvil distintos, como aire, mar y bajo el agua; empezar realizando el hardware necesario para la operación de robots en los ambientes mencionados y hacer que confluyan todos en un sistema computacional, que tenga a la teleoperación como mecanismo de control centralizado sobre cualquiera de los ambientes, manteniendo las capas: física y de desarrollo transparente a los usuarios finales.
8. Implementar una comparación, con el fin de determinar el costo y eficiencia computacional de las plataformas, que emplean el sistema de corrección de errores de odometría con el sistema redundante, con otras técnicas matemáticas como el uso del filtro de Kalman para la corrección de errores; lo que implica la elaboración de un algoritmo que use la plataforma hardware construida y desarrolle el sistema de corrección. Definir la diferencia y ventajas entre los dos sistemas. (fuera de la tesis en el grupo de investigación se está trabajando ya en eso).

7.3 Contribuciones Científicas.

Como se ha mencionado en los capítulos iniciales, este trabajo doctoral, produjo algún material científico divulgado en congresos y revistas internacionales. Es muy importante presentar esto porque a nivel de hardware en robots es muy difícil lograr este tipo de resultados, pero la conjunción de los algoritmos con el hardware altamente desarrollado (en el sentido de su modelación mecánica y de movimiento) hicieron que se vea un interés por lo desarrollado. Es así que se hicieron tres artículos de los cuales dos están publicados y uno en revisión de revista, estos trabajos los cito a continuación:

7.3.1 *Revistas Internacionales con factor de impacto.*

- “Multirobot heterogeneous control considering secondary objectives”, Acosta, J. F., de Rivera, G. G., Andaluz, V. H., & Garrido, J. (2019), *Sensors* (Switzerland), 19(20) doi:10.3390/s19204367.
- “Energy-Saver Mobile Manipulator Based on Numerical Methods”, Acosta Núñez, J.F.; Andaluz Ortiz, V.H.; González-de-Rivera Peces, G.; Garrido Salas, J. (2019) *Electronics* (Switzerland) 2019, 8(10), 1100; <https://doi.org/10.3390/electronics8101100>.
- “Redundant system for the correction of odometry errors for mobile unicycle and omnidirectional robots using artificial vision and genetic algorithms”, Acosta Núñez, J.F.; Andaluz Ortiz, V.H.; González-de-Rivera Peces, G.; Garrido Salas, J. (2019) *Electronics* (Switzerland) (ARTICULO EN REVISION)

7.3.2 *Conferencias Internacionales.*

- “3-D path planning using subatomic particles and feynman diagrams.”, Acosta, J. F., Andaluz, V. H., Naranjo, M. X., Molina, J. I., Santana, A. G., Topa, A. O., & Erazo, G. (2019). Paper presented at the Proceedings of the 3rd World Conference on Smart Trends in Systems, Security and Sustainability, WorldS4 2019, 368-375. doi:10.1109/WorldS4.2019.8903957
- “Path Planning Based in Algorithm Rapidly-Exploring Random Tree RRT”, Asqui, Leonardo; Andaluz, Víctor; Sánchez, Jorge; Acosta, Julio, *Advanced Science Letters*, Volume 24, Number 11, November 2018, pp. 8831-8836(6), American Scientific Publishers DOI: <https://doi.org/10.1166/asl.2018.12356>.
- “Autonomous assistance system for people with amyotrophic lateral sclerosis”, Santana G., A., Ortiz C., O., Acosta, J. F., & Andaluz, V. H. (2017). *Lecture Notes in Electrical Engineering* Volume 449, 2017, Pages 267-277, 7th International Conference on IT Convergence and Security, ICITCS 2017; Seoul; South Korea; 25 September 2017 through 28 September 2017; Code 193979 doi:10.1007/978-981-10-6451-7_32
- “Implementing a multisensory robotic platform for building collaborative monitored robots.”, Acosta, J. F., Glez-de-Rivera, G., & Garrido, J. (2014). Paper presented at the Proceedings of the 2014 29th Conference on Design of Circuits and Integrated Systems, DCIS 2014, doi:10.1109/DCIS.2014.7035592

Como se dice en el mundo de las publicaciones, se tiene a más, cinco artículos en “prensa”, derivados de los resultados obtenidos en este trabajo.

Referencias

- [1] J. Cui, J. T. Wen and J. Trinkle, "A Multi-Sensor Next-Best-View Framework for Geometric Model-Based Robotics Applications," 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 2019, pp. 8769-8775.
- [2] L. Delobel, R. Aufrère, C. Debain, R. Chapuis and T. Chateau, "A Real-Time Map Refinement Method Using a Multi-Sensor Localization Framework," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 5, pp. 1644-1658, May 2019.
- [3] E. Edlerman and R. Linker, "Autonomous Multi-Robot System for use in Vineyards and Orchards," 2019 27th Mediterranean Conference on Control and Automation (MED), Akko, Israel, 2019, pp. 274-279.
- [4] K. Burnett et al., "Building a Winning Self-Driving Car in Six Months," 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 2019, pp. 9583-9589.
- [5] E. Robot, E. Aldousari, M. Alfadhli, A. Alanzi, M. Qasem and R. Noorani, "Design of a Multi Sensor-Based Low-Cost," 2019 International Conference on Robotics,Electrical and Signal Processing Techniques (ICREST), Dhaka, Bangladesh, 2019, pp. 668-670.
- [6] Z. Gong et al., "Design of a Reconfigurable Multi-Sensor Testbed for Autonomous Vehicles and Ground Robots," 2019 IEEE International Symposium on Circuits and Systems (ISCAS), Sapporo, Japan, 2019, pp. 1-5.
- [7] Y. Wu, P. Balatti, M. Lorenzini, F. Zhao, W. Kim and A. Ajoudani, "A Teleoperation Interface for Loco-Manipulation Control of Mobile Collaborative Robotic Assistant," in IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 3593-3600, Oct. 2019.
- [8] J. Ross, J. Lindsay, E. Gregson, A. Moore, J. Patel and M. Seto, "Collaboration of multi-domain marine robots towards above and below-water characterization of floating targets," 2019 IEEE International Symposium on Robotic and Sensors Environments (ROSE), Ottawa, ON, Canada, 2019, pp. 1-7.
- [9] R. Cotsakis, D. St-Onge and G. Beltrame, "Decentralized collaborative transport of fabrics using micro-UAVs," 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 2019, pp. 7734-7740.
- [10] H. Ponce and P. V. C. Souza, "Intelligent Control Navigation Emerging on Multiple Mobile Robots Applying Social Wound Treatment," 2019 IEEE International Parallel and

Distributed Processing Symposium Workshops (IPDPSW), Rio de Janeiro, Brazil, 2019, pp. 559-564.

- [11] M. Turanli and H. Temeltas, "Multi-Robot Collaborative Coverage Under Localization Uncertainty," 2019 IEEE International Conference on Mechatronics and Automation (ICMA), Tianjin, China, 2019, pp. 1999-2005.
- [12] Y. Huang, Y. Zhang and H. Xiao, "Multi-robot system task allocation mechanism for smart factory," 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 2019, pp. 587-591.
- [13] J. Zhu, H. Yu, Z. Lin, N. Liu, H. Sun and M. Liu, "Efficient Actuator Failure Avoidance Mobile Charging for Wireless Sensor and Actuator Networks," in IEEE Access, vol. 7, pp. 104197-104209, 2019.
- [14] T. Xu, P. Yang and H. Guo, "Energy Efficiency Analysis on Robotic Mobile Fulfillment System," 2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA), Tokyo, Japan, 2019, pp. 145-149.
- [15] S. Wang, M. Xia and Y. Wu, "Joint Communication and Motion Energy Minimization in UGV Backscatter Communication," ICC 2019 - 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 2019, pp. 1-6.
- [16] A. Mihály, P. Gáspár and H. Basargan, "Maximizing autonomous in-wheel electric vehicle battery state of charge with optimal control allocation," 2019 18th European Control Conference (ECC), Naples, Italy, 2019, pp. 250-255.
- [17] J. Wang, J. Chen and P. Bao, "Minimum-energy Optimal Control System for Four-wheeled Omni-directional Mobile Robot," 2019 3rd International Symposium on Autonomous Systems (ISAS), Shanghai, China, 2019, pp. 23-28.
- [18] T. A. Sami and S. Momen, "Optimization of Energy Consumption in Swarms of Robots," 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), Agadir, Morocco, 2019, pp. 1-5.
- [19] Y. Liu, L. Zuo, C. Zhang and F. Liu, "Fast and Accurate Robot Localization through Multi-Layer Pose Correction," 2019 IEEE International Conference on Mechatronics and Automation (ICMA), Tianjin, China, 2019, pp. 2487-2492.
- [20] Z. Qiu, Y. Zhuang, F. Yan, H. Hu and W. Wang, "RGB-DI Images and Full Convolution Neural Network-Based Outdoor Scene Understanding for Mobile Robots," in IEEE Transactions on Instrumentation and Measurement, vol. 68, no. 1, pp. 27-37, Jan. 2019.

- [21] V. Peretroukhin and J. Kelly, "DPC-Net: Deep Pose Correction for Visual Localization," in IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 2424-2431, July 2018.
- [22] J. Al Hage, S. Mafrica, M. El Badaoui El Najjar and F. Ruffier, "Informational Framework for Minimalistic Visual Odometry on Outdoor Robot," in IEEE Transactions on Instrumentation and Measurement, vol. 68, no. 8, pp. 2988-2995, Aug. 2019.
- [23] M. Quan, S. Piao, M. Tan and S. Huang, "Accurate Monocular Visual-Inertial SLAM Using a Map-Assisted EKF Approach," in IEEE Access, vol. 7, pp. 34289-34300, 2019.
- [24] E. J. Shamwell, S. Leung and W. D. Nothwang, "Vision-Aided Absolute Trajectory Estimation Using an Unsupervised Deep Network with Online Error Correction," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, 2018, pp. 2524-2531.
- [25] Christine Long, Jason Kridner, BeagleBoard.org; Toda la información se encuentra en su portal web: <http://beagleboard.org/bone-original>
- [26] Caroline Brown, Chris Mair, David Cleavelly, Jon Drori, Kim Shillinglaw, Richard Clegg, Sherry Coutu, Tilly Blyth, Raspberry Pi Foundation, Toda la información se encuentra en su portalweb: <https://www.raspberrypi.org/products/>
- [27] Dual H-Bridge Motor Driver fabricado por TI cuya información se encuentra en el recurso web: <http://www.ti.com/lit/ds/symlink/drv8833.pdf?ts=1588217631138>
- [28] 4-Bit Bidirectional Voltage-Level Shifter with Auto Direction Sensing and +/- 15kV ESD Protect fabricado por TI cuya informacion se encuentra en el recurso web: https://www.ti.com/product/TXB0104?utm_source=google&utm_medium=cpc&utm_campaign=hval-null-null-GPN_EN-cpc-pf-google-ww&utm_content=TXB0104&ds_k=TXB0104&DCM=yes&gclid=EAIaIQobChMIr4uX7ZyP6QIViZOzCh3xZg0rEAAYASAAEgLTRvD_BwE&gclsrc=aw.ds